VULTR

Multi-Region NetApp ONTAP with NetApp FlexCache for Accelerating AI & Analytics on Vultr Cloud with NVIDIA GPU

Table of Contents

Introduction

Organizations can now seamlessly extend their on-premises NetApp ONTAP environments into Vultr Cloud, enabling AI and analytics workloads to run with dramatically lower data-access latency. Using NetApp FlexCache, customers present their existing datasets in Vultr as secure, read/write volumes - without migrating or duplicating data. This approach places business-critical data closer to NVIDIA GPUs in Vultr Cloud, accelerating AI workloads, reducing infrastructure costs, and enabling rapid experimentation across regions and teams.

NetApp FlexCache creates lightweight, high-performance cache volumes in Vultr Cloud that automatically fetch frequently accessed data from on-premises ONTAP systems while synchronizing all writes to the source. This ensures consistent, real-time access to enterprise datasets across sites and clouds, without the overhead of replicating full datasets. The result is a unified hybrid-cloud architecture that supports high-performance AI, ML, and analytics workloads on NVIDIA GPUs, while preserving data governance, security, and operational simplicity.

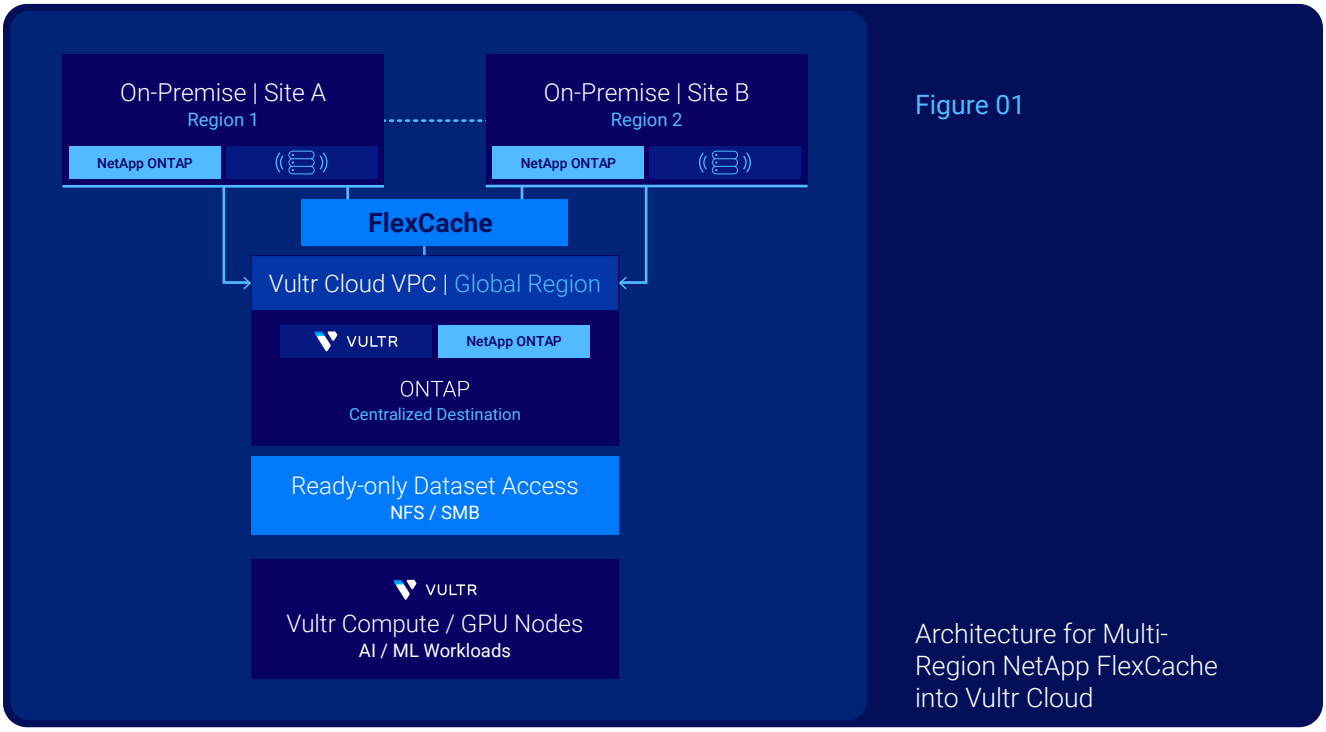


Figure 01

Architecture for Multi-Region NetApp FlexCache into Vultr Cloud

This hybrid-cloud design extends on-premises NetApp ONTAP data into Vultr Cloud using NetApp FlexCache, giving GPU workloads fast, local access to enterprise datasets without migrating full volumes. The result is a unified, low-latency data layer optimized for AI and analytics.

Hybrid Multi-Region Data Consolidation:

- **NetApp FlexCache fan-in** brings data from multiple ONTAP sites into a single ONTAP instance on Vultr.
- **Standard NFS/SMB access** lets Vultr compute mount datasets immediately.
- **Supports ONTAP, CVO, and AFF/FAS.**

NetApp FlexCache Performance & Efficiency:

- **Low-latency access** near Vultr GPU nodes.
- **Automatic caching** of hot data.
- **Real-time coherency** with on-prem origin volumes.
- **Small cloud footprint** since only active data is cached.
- **Scales easily** to many sites.

Enterprise-Ready Hybrid Cloud Design:

- **Aligned with NetApp best practices.**
- **Optimized for AI/ML** by placing data close to compute.

This solution consolidates data from multiple on-premises ONTAP environments into a unified hybrid-cloud platform on Vultr. Each site exposes its active data as a NetApp FlexCache volume, giving compute and GPU workloads consistent, region-wide access without shifting primary datasets off-premises.

NetApp FlexCache further enhances performance by positioning lightweight cache volumes near consuming applications. Hot data is fetched on demand, writes are synchronized back to the origin, and full replication is avoided - resulting in a low-latency, efficient, and highly scalable data layer for multi-region AI and enterprise workloads.

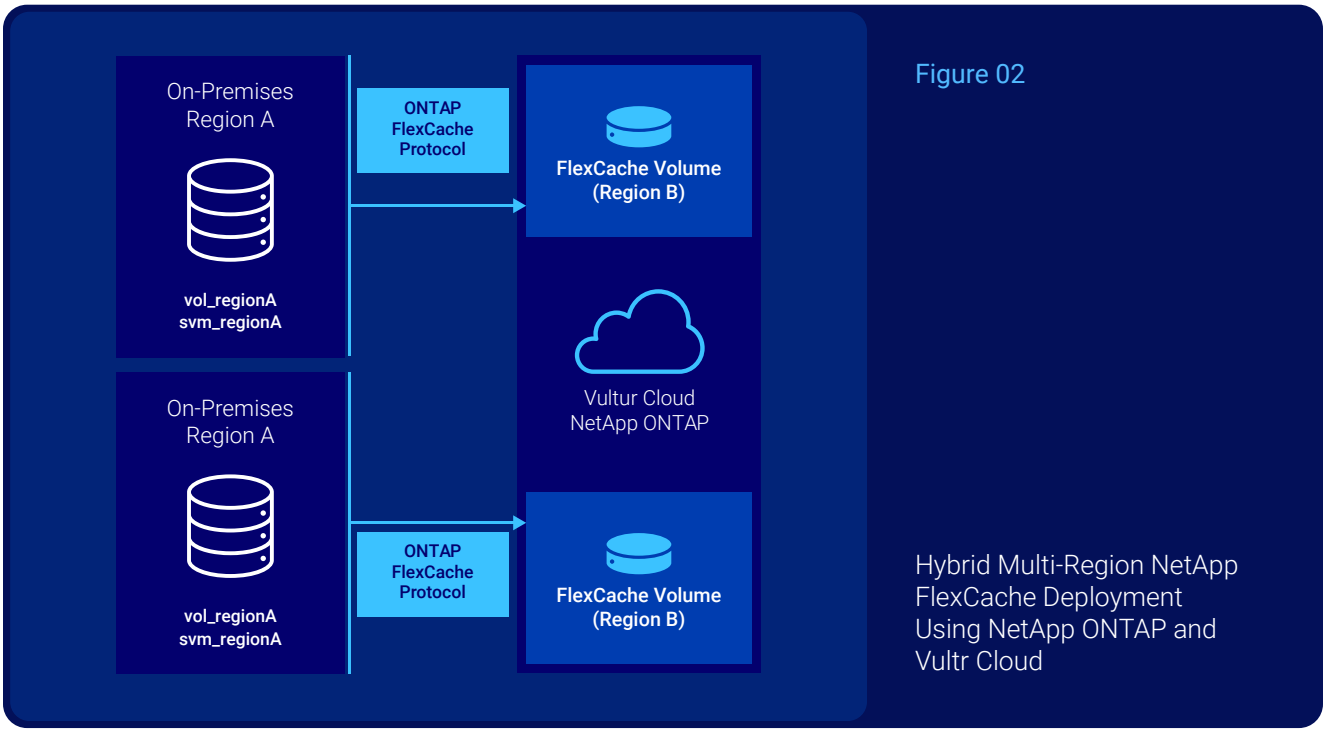


Figure 02

Hybrid Multi-Region NetApp FlexCache Deployment Using NetApp ONTAP and Vultr Cloud

Prerequisites

The prerequisites below ensure that the environment is properly configured for a multi-region NetApp FlexCache setup using NetApp ONTAP on Vultr Cloud as the destination.

ONTAP

- ONTAP 9.8+ on all source and destination clusters
- NetApp ONTAP requires matching major ONTAP version on both Source and destination
- NetApp FlexCache license on both origin and cache clusters
- Intercluster LIFs configured on each node
- DNS, routing, and firewall rules in place
- Source SVMs and volumes ready for replication

ONTAP on VMware ESXi (in Vultr Cloud)

- Vultr-provided ESXi 8.x or ESXi 9.0 host with vCenter access to deploy given ONTAP ova file, create/assign datastores, and create port groups / networks
- ONTAP license available to be deployed on ESXi in Vultr Cloud
- Adequate CPU/RAM and attached block storage
- Management, inter-cluster, and data LIFs configured.

Networking

- IPSec VPN or secure tunnel between sites and Vultr
- Connectivity between intercluster LIFs (ping + TCP 11104/11105/10000)
- NFS/SMB ports allowed for compute access
- DNS resolution across environments

Security & Access

- Admin access to all ONTAP clusters
- Vultr console/API access
- Firewall rules permitting cluster/SVM peering for NetApp FlexCache

Storage & Configuration

- Destination aggregates created on NetApp ONTAP
- SVMs defined for replication/caching and data access boundaries

Step 1 - Prepare Networking & Environment

A secure, stable network foundation is required before configuring NetApp Flexcache. That includes creating the VPC, deploying a VPN tunnel, setting up DNS and routing, and deploying NetApp ONTAP in Vultr Cloud. NetApp FlexCache depends on encrypted TCP connectivity between intercluster LIFs - provided by the VPN - so correct resource provisioning and network configuration are critical to ensure all downstream steps work smoothly without impacting production.

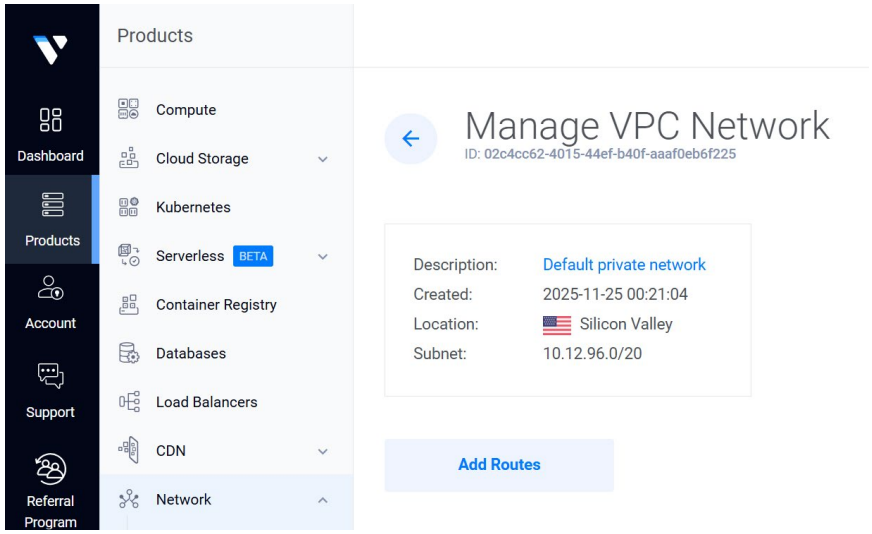
This section covers

- Create and validate Vultr Cloud account
- Create Vultr VPC and required compute instances
- Set up IPSec VPN between on-prem ONTAP and Vultr Cloud
- Validate routing using ICMP, NFS ports, and NetApp FlexCache ports (TCP 11104/11105/10000)
- Deploy ONTAP image in Vultr
- Create ONTAP SVM (vserver) and configure DNS, routing, and firewall rules

Create Vultr VPC

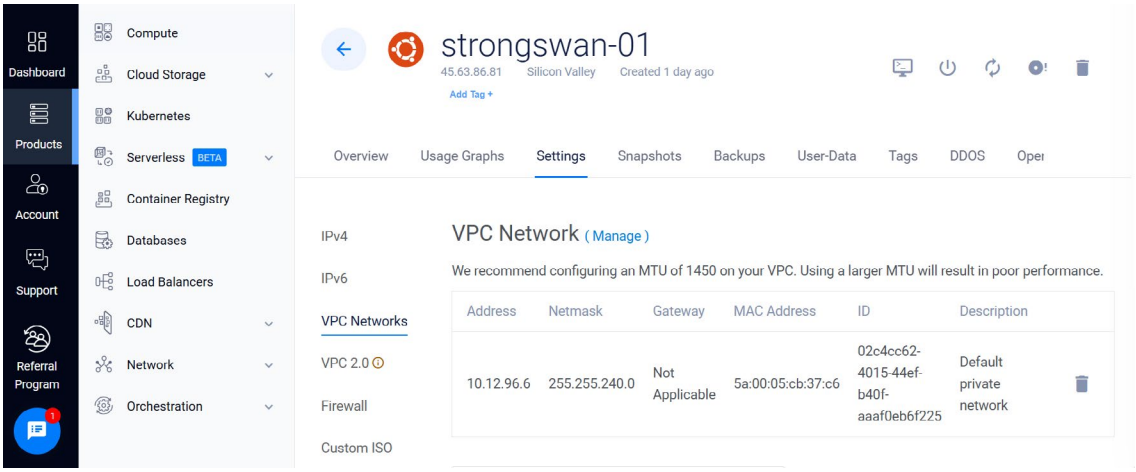
A Vultr VPC is a logically isolated virtual network that provides private IP addressing and controlled routing for cloud resources. It is needed to securely host ONTAP and compute nodes in an isolated environment, ensuring predictable connectivity and a dedicated path for NetApp FlexCache traffic.

Provision a VPC to provide network isolation and private IP space for ONTAP and compute nodes.



Expected Output:

- The VPC should display the correct subnet 10.xx.xx.0/20 and region.
- This confirms the VPC was created and provides the private network required for ONTAP and NetApp FlexCache traffic.



Expected Output:

- The StrongSwan VM should have a private VPC IP address (e.g., 10.xx.xx.xx).
- This confirms the VM is attached to the correct Vultr VPC and can communicate with ONTAP.

Deploy ONTAP on VMware ESXi

Deploy the ONTAP VM that will serve as the NetApp FlexCache destination. This ONTAP instance provides full ONTAP capabilities in Vultr Cloud, enabling FlexCache and exposing NFS/SMB volumes to Vultr compute nodes.

- Log in to your **VMware ESXi host.**
- Go to **Networking** » **Virtual Switches.**
- Create a new virtual switch named **vSwitch1** and click **Add**. This switch will be used for the VPC connection.
- Select **Port Groups** » **Add Port Group.**
- Enter a **Port Group Name,** choose **vSwitch1** as the virtual switch, and click **Add.**
- In the left pane, select **Virtual Machines** and click **Create / Register VM.**
- Choose **Deploy a virtual machine** from an OVA file, then click Next.
- Provide a **VM Name,** select the **OVA file,** and click **Next.**
- On the **Storage Options** page, click **Next** to continue.
- Under **Deployment Options,** select your VM Network, then click **Next.**
- On **Additional Settings,** enter the admin password and network configuration details, then click **Next** » **Finish.**
- The deployment may take 4–5 minutes, depending on your network speed.
- Once the login screen appears in the VM console, open a browser and navigate to the configured IP address to access the **ONTAP Deploy Dashboard.**

Deploy ONTAP System Manager Cluster

- Log in to the **ONTAP Deploy Dashboard.**
- **Upload your license file.**
- Under **Add Host to Inventory,** choose **Hypervisor Type » ESX**, enter your V**Mware ESXi credentials,** and click **Add.**
- Under **Create a Cluster,** provide a **Cluster Name,** select a **Cluster Size,** and choose the appropriate **Network Configuration,** then click Done.
- In Hypervisor and Network, configure:
  - Node settings
  - VMware hosts
  - Required networks
- In **Storage,** select **Software RAID**. Under **Storage Pool,** ensure **all disks are the same size** for optimal stability. Using mismatched disks can cause configuration failures, VM restarts, or cluster instability.
- Click **Done** to proceed.
- Set the **admin password** for **ONTAP System Manager** and click **Create Cluster.**
- The cluster deployment will take about **4–5 minutes.**
- Return to the **VMware ESXi** panel and wait for the cluster VM to finish deploying.
- When the login page appears on the cluster VM console, go back to the **ONTAP Deploy Dashboard.** Under Clusters, select your cluster and click **Launch System Manager.**
- Log in with your credentials to access and manage storage through the **System Manager interface.**

Create LIF on NetApp ONTAP

An **Inter-cluster LIF on NetApp ONTAP** is a dedicated logical network interface assigned a private IP inside the VPN/VPC subnet and bound to a specific node and port. It serves as the endpoint for all cluster-to-cluster communication, including SnapMirror replication, NetApp FlexCache and cluster peering. ONTAP does not create these automatically, so defining at least one inter-cluster LIF per node is essential - without it, the cluster cannot establish peering or replicate data, even if the VPN tunnel is fully functional.

Check the available ports:

```
net port show

ONTAPSelectCluster::> net port show
  (network port show)

Node: ONTAPSelectCluster-01
                                                  Speed(Mbps) Health
Port      IPspace      Broadcast Domain Link MTU  Admin/Oper  Status
--------- ------------ ---------------- ---- ---- ----------- --------
e0a       Default      Default          up   1500  auto/auto  healthy
e0b       Default      Default          up   1500  auto/auto  healthy
e0c       Default      Default          up   1500  auto/auto  healthy
3 entries were displayed.

ONTAPSelectCluster::>
```

Use e0a which is the **default, stable, and recommended data-capable port** available on ONTAP for intercluster/LIF traffic in most deployments.


```
net int create -vserver ONTAPSelectCluster -lif ic1 -address 10.12.96.51 -netmask
255.255.240.0 -home-node ONTAPSelectCluster-01 -home-port e0a –role intercluster
```

```
ONTAPSelectCluster::> net int create -vserver ONTAPSelectCluster -lif ic1 -address 10.12.96.51 -netmask 255.255.240.0 -home-node ONTA
PSelectCluster-01 -home-port e0a
```

CLI Validation

Check LIF state:

```
network interface show

ONTAPSelectCluster::> network interface show
            Logical    Status     Network            Current       Current Is
Vserver     Interface  Admin/Oper Address/Mask       Node          Port    Home
----------- ---------- ---------- ------------------ ------------- ------- ----
ONTAPSelectCluster
            ONTAPSelectCluster-01_mgmt1 up/up 10.12.96.61/20 ONTAPSelectCluster-01 e0a true
            cluster_mgmt up/up    10.12.96.60/20     ONTAPSelectCluster-01 e0a true
            ic1          up/up    10.12.96.51/20     ONTAPSelectCluster-01 e0a true
SVM_Dataflex
            SVM_Dataflex_data_01 up/up 10.12.96.53/20 ONTAPSelectCluster-01 e0b true
SVM_data
            SVM_data_Cifs up/up   10.12.96.52/20     ONTAPSelectCluster-01 e0c true
5 entries were displayed.
```

```
net int show -role intercluster -fields role,address,status-oper,
status-admin,lif,home-port,home-node
```

```
ONTAPSelectCluster::> net int show -role intercluster -fields role,address,status-oper,status-admin,lif,home-port,home-node
  (network interface show)
vserver           lif role         address     home-node             home-port status-oper status-admin
----------------- --- ------------ ----------- --------------------- --------- ----------- ------------
ONTAPSelectCluster ic1 intercluster 10.12.96.51 ONTAPSelectCluster-01 e0a       up          up
```

Expected Output:

- lif: ic1
- Address: 10.12.96.51
- Home Node: ONTAPSelectCluster-01
- Current Node: ONTAPSelectCluster-01 (unless it has failed over)
- Home Port: **e0a**
- Operational Status: up
- Administrative Status: up

Enable NFS on ONTAP

Enable NFS using the following command on NetApp ONTAP

```
vserver nfs create -vserver <vservername> -v3 enabled -v4.0 enabled
```

CLI (Region A)

```
vserver nfs create -vserver SVM_data -v3 enabled -v4.0 enabled
```


Verify NFS server creating with:

```
nfs show
ONTAPSelectCluster::> vserver nfs create -vserver SVM_data -v3 enabled -v4.0 enabled

ONTAPSelectCluster::> nfs show

Vserver: SVM_data

        General Access:  true
                    v3:  enabled
                  v4.0:  enabled
                   4.1:  enabled
                   UDP:  enabled
                   TCP:  enabled
                  RDMA:  enabled
  Default Windows User:  -
 Default Windows Group:  -
```

CLI (Region B)

```
nfs create -vserver SVM_Dataflex -v3 enabled -v4.0 enabled
```

Verify NFS server creating with:

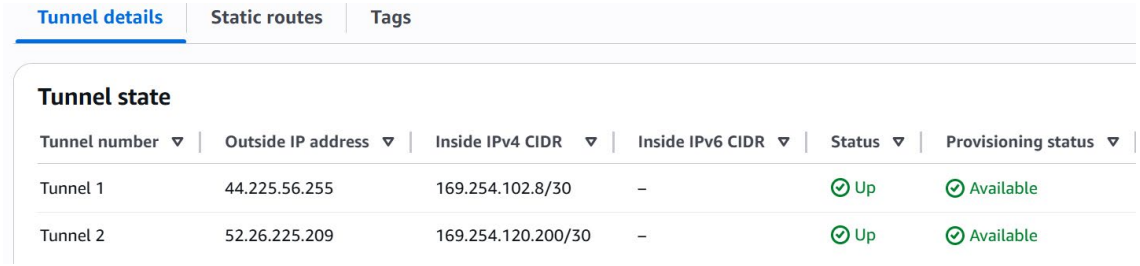
```
nfs show

ONTAPSelectCluster::> nfs show

Vserver: SVM_Dataflex

        General Access:  true
                    v3:  enabled
                  v4.0:  enabled
                   4.1:  enabled
                   UDP:  enabled
                   TCP:  enabled
                  RDMA:  enabled
  Default Windows User:  -
 Default Windows Group:  -
```

Set Up VPN/IPSec

NetApp FlexCache replication requires secure, encrypted TCP connectivity between intercluster LIFs. A VPN/IPSec tunnel provides that secure transport path across public networks. Set up a secure IPSec tunnel between on-prem ONTAP and Vultr Cloud.

The VPN tunnel provides the **secure transport path** between the on-prem and Vultr environments.Without it, your ONTAP VM inside Vultr has **no route to the on-prem gateway or DNS** for name resolution or reachability tests.

StrongSwan was deployed on the Vultr compute node to serve as the VPN end-point for this build guide.




```
ipsec statusall
```

Expected Output:

- Both IPSec tunnels should show **Status: UP.**

This confirms On-prem has successfully established Phase-1 and Phase-2 security associations with the StrongSwan gateway in Vultr.

Configure DNS & Routing

After the VPN is active, you can safely set default and inter-cluster routes so ONTAP can:

- Resolve hostnames across sites
- Reach on-prem inter-cluster LIFs over the new encrypted tunnel

Steps:

- Add DNS servers
- Add a default route pointing through the VPN gateway
- Add inter-cluster routes for the on-prem CIDRs

Set DNS servers and create default/inter-cluster routes

Add local host entries for peer clusters

- 10.12.96.51 » **Vultr ONTAP intercluster LIF IP**
- `172.31.157.201` » **On-Prem ONTAP intercluster LIF IP**

These private IPs inside the VPN tunnel are the internal intercluster LIF addresses used exclusively for SnapMirror replication and NetApp FlexCache. ONTAP relies on these dedicated logical interfaces for all replication traffic.

Note: They are *not* the public IPs of StrongSwan or bastion hosts.
They are the *internal* cluster communication IPs used for NetApp FlexCache.

Each ONTAP (On-Prem and Vultr) must be able to:

- Resolve the **peer cluster's intercluster LIF hostname**
- Reach that IP across the VPN (on TCP ports 11104 and 11105)

On Vultr ONTAP

CLI

```
vserver services name-service dns hosts create -vserver ONTAPSelectCluster -address
172.31.157.201 -hostname inter_1
```

Verify:

```
vserver services name-service dns show
```

```
ONTAPSelectCluster::> vserver services name-service dns hosts show
Vserver    Address        Hostname        Aliases
---------- -------------- --------------- ----------------------
ONTAPSelectCluster
           172.31.157.201 inter_1         -
```

CLI on On-Prem ONTAP

```
vserver services name-service dns hosts create -vserver sx -address 10.12.96.51
-hostname ic1
```

Verify:

```
vserver services name-service dns hosts show
```

```
sxId0edb1927eae795ba7::> vserver services name-service dns hosts show

Vserver     Address        Hostname        Aliases

----------  -------------- --------------- ----------------------

sx          10.12.96.51    ic1
```

Create Route

This is the Vultr ONTAP VPC subnet: 10.12.96.0/20

- ONTAP intercluster LIF: 10.12.96.51
- StrongSWAN LAN IP (VPN gateway in Vultr): 10.12.96.6

On-Premise ONTAP VPC subnet: 172.31.0.0/16

- On-Prem ONTAP Intercluster LIF: 172.31.157.201
- On-Prem local subnet gateway: 172.31.144.1

Note: The routes are shown for the IPs above. This would change as per setup.

Route to be added on Vultr ONTAP (to reach on-prem equivalent)

To reach anything in the On-Prem ONTAP VPC (172.31.0.0/16), send traffic to the StrongSWAN LAN IP (10.12.96.6), which is the VPN gateway on the Vultr side.

```
network route create -vserver ONTAPSelectCluster -destination 172.31.0.0/16 -gateway
10.12.96.6
```

CLI Validation

```
network route show

ONTAPSelectCluster::> network route show
Vserver             Destination     Gateway         Metric
------------------- --------------- --------------- ------
ONTAPSelectCluster
                    0.0.0.0/0       10.12.96.1      10
                    172.31.0.0/16   10.12.96.6      20
2 entries were displayed.

ONTAPSelectCluster::>
```

Expected Output:

- A default route exists
- Specific route(s) to on-prem network appear
- No overlapping or incorrect routes

Route to be added on On-Prem ONTAP (to reach Vultr ONTAP)

To reach the Vultr ONTAP subnet (10.12.96.0/20), send traffic to On-Prem local VPC subnet gateway (172.31.144.1)

```
network route create -vserver sx -destination 10.12.96.0/20 -gateway 172.31.144.1
```

CLI Validation

```
network route show
```

```
sxId0edb1927eae795ba7::*> network route show
 Vserver             Destination     Gateway         Metric
 ------------------- --------------- --------------- ------
 sx
                     0.0.0.0/0       172.31.144.1    20
                     10.12.96.0/20   172.31.144.1    20
 2 entries were displayed.
```

Validate Connectivity

To validate that the VPN tunnel and routing are correctly configured for NetApp FlexCache, both sites must be able to reach each other's **intercluster LIFs**. ONTAP provides a built-in way to test this using network ping **from the LIF itself**, ensuring that the test follows the correct broadcast domain, routing table, and VPN path.

CLI Validation

Test From On-Prem » ONTAP (Cloud)

```
network ping -lif inter_1 -vserver sxId0edb1927eae795ba7 -destination <Cloud_IC_LIF_IP>

network ping -lif inter_2 -vserver sxId0edb1927eae795ba7 -destination <Cloud_IC_LIF_IP>
```

Use IP address and not ic1. ic1 is the *name* of a LIF object inside ONTAP, not a DNS hostname, so ONTAP can't resolve it unless we explicitly create a DNS/hosts entry with that name.

```
sxId0edb1927eae795ba7::> network ping -lif inter_1 -vserver
sxId0edb1927eae795ba7 -destination 10.12.96.51

 10.12.96.51 is alive

sxId0edb1927eae795ba7::> network ping -lif inter_2 -vserver
sxId0edb1927eae795ba7 -destination 10.12.96.51

 10.12.96.51 is alive
```

Test From ONTAP (Cloud) » On-Prem

```
network ping -lif ic1 -vserver ONTAPSelectCluster -destination <ONprem_IC_
LIF_1>

ONTAPSelectCluster::> network ping -lif ic1 -vserver ONTAPSelectCluster
-destination 172.31.157.201

 172.31.157.201 is alive

network ping -lif ic1 -vserver ONTAPSelectCluster -destination <ONprem_IC_
LIF_2>

ONTAPSelectCluster::> network ping -lif ic1 -vserver ONTAPSelectCluster
-destination 172.31.144.104

 172.31.144.104 is alive
```

Expected Output:

- **<OnPrem_IP>** is alive
- Symmetric reachability
- No routing errors (e.g., "Network unreachable")

This validates that the ONTAP Vultr Cloud intercluster LIF can return traffic back to the on-prem intercluster LIFs.

NetApp FlexCache Firewall Ports

The following TCP ports **must be enabled bidirectionally** on your firewall between the Intercluster LIFs of your ONTAP clusters (On-Prem and Vultr ONTAP) before cluster peering can be established.

- **TCP Port 11104 (SnapMirror and NetApp FlexCache Control)**
  - **Purpose:** Mandatory for establishing and maintaining cluster peering.
  - **Function:** Used for the secure control channel, authentication, and metadata exchange (the peering handshake).
- **TCP Port 10000 (Data Transfer Primarily for SnapMirror)**
  - **Purpose:** Mandatory for all SnapMirror data transfer operations only
  - **Function:** Used for the actual block-level data transfer during replication.

CLI run on On-Prem:

```
nc -zv 10.12.96.51 11104
nc -zv 10.12.96.51 11105
```

```
ubuntu@ip-172-31-12-99:~$ nc -zv 10.12.96.51 11104
Connection to 10.12.96.51 11104 port [tcp/*] succeeded!
ubuntu@ip-172-31-12-99:~$ nc -zv 10.12.96.51 11105
Connection to 10.12.96.51 11105 port [tcp/*] succeeded!
ubuntu@ip-172-31-12-99:~$ nc -zv 10.12.96.51 10000
Connection to 10.12.96.51 10000 port [tcp/webmin] succeeded!
```

CLI run on ONTAP:

```
nc -zv 172.31.157.201 11104
nc -zv 172.31.157.201 11105
```

```
root@strongswan-01:~# nc -zv 172.31.157.201 11104
Connection to 172.31.157.201 11104 port [tcp/*] succeeded!
root@strongswan-01:~# nc -zv 172.31.157.201 11105
Connection to 172.31.157.201 11105 port [tcp/*] succeeded!
root@strongswan-01:~# nc -zv 172.31.157.201 10000
Connection to 172.31.157.201 10000 port [tcp/webmin] succeeded!
```

Expected Output:

- Connection to ... port [tcp/*] succeeded!
- Zero connection timeouts or refusals.
- Confirms bidirectional traffic flow on critical ports.

Essential NFS Access Ports

These ports are necessary for your Vultr Compute/GPU Nodes to successfully mount and access the read-only datasets from the NetApp ONTAP system. They must be opened on the firewall governing traffic between the compute nodes and the ONTAP Data LIFs.

- **TCP/UDP Port 111**
  - **Purpose:** Mandatory for initial client-server communication and service discovery.
  - **Function:** Used by the NFS client (Vultr Compute Nodes) to find the dynamic port numbers of other necessary NFS services, such as Mountd.
- **TCP/UDP Port 2049**
  - **Purpose:** Mandatory for all NFS data access and transfer operations.
  - **Function:** The primary port used for the core NFS protocol, enabling clients to read and write file data across the network. (Note: For **NFSv4.x**, this port often handles all functions.)

CLI run on ONTAP:

To check if a **Vultr Compute Node** can reach the **NetApp ONTAP Data LIF** (10.12.96.52) on the main NFS port (2049):

Get NetApp ONTAP Data LIF

```
network interface show –role data

ONTAPSelectCluster::*> net int show -role data
  (network interface show)
           Logical    Status     Network            Current       Current Is
Vserver    Interface  Admin/Oper Address/Mask       Node          Port    Home
---------- ---------- ---------- ------------------ ------------- ------- ----
SVM_Dataflex
           SVM_Dataflex_data_01 up/up 10.12.96.53/20 ONTAPSelectCluster-01 e0b true
SVM_data
           SVM_data_Cifs up/up   10.12.96.52/20     ONTAPSelectCluster-01 e0c true
2 entries were displayed.
```

Use the above ONTAP Data LIF to run the netcat command.

```
nc -zv 10.12.96.52 2049
nc -zv 10.12.96.52 111

root@strongswan-01:~# nc -zv 10.12.96.52 2049
Connection to 10.12.96.52 2049 port [tcp/nfs] succeeded!
root@strongswan-01:~# nc -zv 10.12.96.52 111
Connection to 10.12.96.52 111 port [tcp/sunrpc] succeeded!
```

```
nc -zv 10.12.96.53 2049
nc -zv 10.12.96.53 111

root@strongswan-01:~# nc -zv 10.12.96.53 2049
Connection to 10.12.96.53 2049 port [tcp/nfs] succeeded!
root@strongswan-01:~# nc -zv 10.12.96.53 111
Connection to 10.12.96.53 111 port [tcp/sunrpc] succeeded!
```

Step 2 – Configure ONTAP Cluster Peering

Cluster and SVM peering establish the trust required for NetApp FlexCache relationships across on-premises and Vultr ONTAP environments. Peering allows the systems to communicate securely over intercluster LIFs and authorizes the cache to access the origin volume. Without proper peering in place, a NetApp FlexCache volume cannot be created or connected to its source.

This section covers:

- Create intercluster LIFs on both on-prem ONTAP and Vultr NetApp ONTAP
- Configure cluster peering (bidirectional trust)
- Configure SVM peering (data SVM to data SVM authorization)
- Validate encrypted intercluster connectivity

Cluster Peering

Cluster peering is essential for enabling NetApp FlexCache. It establishes a secure trust relationship between source and destination ONTAP clusters, allowing them to authenticate and exchange replication metadata. This process relies entirely on **Inter-cluster LIFs**, which are dedicated network interfaces used solely for managing the ONTAP-to-ONTAP traffic required for peering and data transfer.

Note: Always run the 'cluster peer create' command first on the destination side - the side that will RECEIVE the peer request.

On NetApp ONTAP on Vultr Cloud

```
network interface show -role intercluster

ONTAPSelectCluster::> network interface show -role intercluster
            Logical    Status     Network            Current       Current Is
Vserver     Interface  Admin/Oper Address/Mask       Node          Port    Home
----------- ---------- ---------- ------------------ ------------- ------- ----
ONTAPSelectCluster
            ic1          up/up    10.12.96.51/20     ONTAPSelectCluster-01 e0a true
```

On On-Prem ONTAP

```
sxId0edb1927eae795ba7::> network interface show -role intercluster
           Logical     Status        Network            Current      Current   Is
Vserver    Interface  Admin/Oper     Address/Mask       Node         Port      Home
---------- ---------- -------------  ------------------ ----------  ---------- -----
sxId0edb1927eae795ba7

                      inter_1      up/up          172.31.157.201/20
sxId0edb1927eae795ba7-01   e0e     true

                      inter_2      up/up          172.31.144.104/20
sxId0edb1927eae795ba7-02   e0e     true
 2 entries were displayed.
```

Expected Output

- status-admin = up, status-oper = up
- Correct IPs assigned
- Home-node and home-port match the configuration

CLI for cluster peering to be run on Vultr Cloud NetApp ONTAP:

```
cluster peer create -peer-addrs <peer-intercluster-LIF-IP of ON-PREM> -generate-passphrase true
```

```
ONTAPSelectCluster::*> cluster peer show

This table is currently empty.

ONTAPSelectCluster::*> cluster peer create -peer-addrs 172.31.144.104 -generate-passphrase true

Notice:

        Passphrase: tl++CrpFI1Y/Axxp+3YzlU+R

        Expiration Time: 12/3/2025 06:37:24 +00:00

        Initial Allowed Vserver Peers: -

        Intercluster LIF IP: 10.12.96.51

        Peer Cluster Name: sxId0edb1927eae795ba7

      Warning: make a note of the passphrase - it cannot be displayed again.
```

Note: Copy the Passphrase which will be needed in the next command

CLI for cluster peering to be run on On-Prem NetApp ONTAP:

Here, don't pass '-generate-passphrase true', as we need to use the generated passphrase from NetApp ONTAP[Destination]

- **Passphrase is generated only once** - on the destination cluster that initiates the peering (Destination NetApp ONTAP).
- The **on-prem source** must **reuse** that same passphrase when replying to complete the peering.

```
cluster peer create -peer-addrs <peer-intercluster-LIF-IP of ON-PREM>


sxId0edb1927eae795ba7::> cluster peer create -address-family ipv4 -peer-addrs 10.12.96.51

Notice: Use a generated passphrase or choose a passphrase of 8 or more
characters. To ensure the authenticity of the peering relationship, use a
phrase or sequence of characters that would be hard to guess.

Enter the passphrase:

Confirm the passphrase:
```

Validation

```
cluster peer show
```

```
ONTAPSelectCluster::*> cluster peer show

Peer Cluster Name         Cluster Serial Number  Availability   Authentication

------------------------- ---------------------  -------------  -----------

sxId0edb1927eae795ba7     1-80-000011            Available       ok


sxId0edb1927eae795ba7::> cluster peer show

Peer Cluster Name       Cluster Serial Number  Availability    Authentication

----------------------- ---------------------  -------------  --------------

ONTAPSelectCluster      1-80-000011            Available       ok
```

Expected Output

- Availability = **Available**
- Authentication = **ok**
- Remote cluster name displayed
- No timeout or “unreachable” errors

SVM Peering

SVM (vserver) peering links the data SVMs on each cluster so NetApp FlexCache can access the origin volume and establish a cache relationship. Because FlexCache authorization is defined at the SVM level, this step is required. Even if the clusters are peered, FlexCache cannot operate unless the source and destination SVMs are explicitly permitted to communicate.

Since the Vultr Cloud ONTAP system will act as the FlexCache destination, it is the appropriate place to initiate the ‘vserver peer create’ command. The on-premises ONTAP system will then receive and accept the peering request.

CLI run on NetApp ONTAP (Vultr Cloud)

Run the following command on the ONTAP cluster (the destination). This command creates the pending peering relationship.

```
vserver peer create -vserver <dst_svm_ontap_select> -peer-vserver <src_svm_onprem>
–applications flexcache,snapmirror –peer-cluster <onprem-cluster-name>
```

```
ONTAPSelectCluster::*> vserver peer create -vserver SVM_data -peer-vserver sx
-applications flexcache,snapmirror -peer-cluster sxId0edb1927eae795ba7

Info: [Job 52] ‘vserver peer create’ job queued
```


CLI run on On-prem NetApp ONTAP [Source]

Immediately after the create command, run the following command on the corresponding **On-Prem cluster** (the source) to accept the peering request and finalize the relationship.

```
server peer accept -vserver <src_svm_onprem> -peer-vserver <dst_svm_ontap_select>
```

```
sxId0edb1927eae795ba7::> vserver peer accept -vserver sx -peer-vserver SVM_data

Info: [Job 82] 'vserver peer accept' job queued
```

Validation from both NetApp ONTAP(Destination) and On-prem ONTAP(Source)

Validation:

```
vserver peer show

ONTAPSelectCluster::*> vserver peer show
            Peer        Peer                           Peering        Remote
Vserver     Vserver     State        Peer Cluster      Applications   Vserver
----------- ----------- ------------ ----------------- -------------- ---------
SVM_data    sx          peered       sxId0edb1927eae795ba7
                                                        flexcache, snapmirror
```

```
sxId0edb1927eae795ba7::> vserver peer show
            Peer        Peer                           Peering        Remote
Vserver     Vserver     State        Peer Cluster      Applications   Vserver
----------- ----------- ------------ ----------------- -------------- ---------
sx          SVM_data    peered       ONTAPSelectCluster
                                                       flexcache, snapmirror
                                                                      SVM_data
```

Expected Output

- Peer state = **peered**
- Applications = **FlexCache, snapmirror**
- No "initial" or "pending" states
- No peer conflicts

Step 3 - Setup Destination Volumes & NetApp FlexCache

NetApp FlexCache enables low-latency, read-optimized access to shared datasets by creating cache volumes close to applications and users while maintaining a single authoritative origin volume. In this workflow, FlexCache volumes are created on the destination ONTAP system to locally serve read traffic, and appropriate NFS/SMB export and security policies are applied to ensure controlled, transparent access for clients. Together, these steps allow applications to consume cached data seamlessly without modifying existing access patterns or duplicating full datasets.

This section covers

- Network Test-Path (NetApp FlexCache Connectivity Validation)
- Prepare destination aggregates
- Configure volume export/security settings (NFS/SMB)
- Create destination NetApp FlexCache volume
- Test mounts on Vultr compute nodes

Network Test-Path (NetApp FlexCache Connectivity Validation)

The network test-path command is a specialized **NetApp ONTAP utility** used to **proactively validate the entire network path** necessary for NetApp FlexCache and cluster peering operations. It is the most robust way to verify your network is ready.

This command should be run **after** Cluster Peering and vserver peering is set up but **before** you attempt to set up NetApp FlexCache.

This tests the following:

- **Connectivity & Routing:** Confirms that a dedicated SnapMirror and NetApp FlexCache connection can be established between the source and destination cluster nodes over the correct **Intercluster LIFs**.
- **Firewall Status:** Verifies that the firewall allows traffic for both the **SnapMirror and NetApp FlexCache control channel (TCP 11104)** and the **data transfer channel (TCP 10000).**
- **LIF Configuration:** Ensures the Intercluster LIFs on both the source and destination are properly configured, up, and listening for SnapMirror and NetApp FlexCache traffic.

```
network test-path -source-node <Source_Node_Name> -destination-cluster <Destination_
Cluster_Name> -destination-node <Destination_Node_Name>
```

```
sxId0edb1927eae795ba7::*> network test-path -source-node sxId0edb1927eae795ba7-01
-destination-cluster ONTAPSelectCluster -destination-node ONTAPSelectCluster-01

Warning: This operation will generate large amount of cluster traffic and can cause
temporary cluster traffic slowness.

Do you want to continue? {y|n}: y

 Initiating path test. It can take up to 120 seconds for results to be displayed.

     Test Duration: 14.25 secs

   Send Throughput: 27.97 MB/sec

Receive Throughput: 27.97 MB/sec

           MB Sent: 398.62

       MB Received: 398.62

       Avg Latency: 5157.99 ms

sxId0edb1927eae795ba7::*> network test-path -source-node sxId0edb1927eae795ba7-02
-destination-cluster ONTAPSelectCluster -destination-node ONTAPSelectCluster-01

Warning: This operation will generate large amount of cluster traffic and can cause
temporary cluster traffic slowness.
```

```
Do you want to continue? {y|n}: y

Test Duration: 14.26 secs

Send Throughput: 29.64 MB/sec

Receive Throughput: 29.64 MB/sec

MB Sent: 422.56

MB Received: 422.56

Avg Latency: 4797.68 ms
```

```
ONTAPSelectCluster::*> network test-path -source-node ONTAPSelectCluster-01
-destination-cluster  sxId0edb1927eae795ba7 -destination-node
sxId0edb1927eae795ba7-01

 Warning: This operation will generate large amount of cluster traffic and can cause
temporary cluster traffic slowness.

Do you want to continue? {y|n}: y

Initiating path test. It can take up to 120 seconds for results to be displayed.

     Test Duration: 14.23 secs

   Send Throughput: 20.71 MB/sec

Receive Throughput: 20.71 MB/sec

           MB Sent: 294.69

       MB Received: 294.69

       Avg Latency: 5387.71 ms
```

Expected Output

- Confirms network connectivity between source and destination nodes
- Shows source node and **destination node/cluster**
- Displays **status of the path** (e.g., success or failure)
- Provides **latency** or **response time** details (if applicable)
- Indicates any issues such as packet loss or unreachable network

Prepare Destination Aggregates

Aggregates are storage pools, and FlexVols are the ONTAP volumes that will receive replicated NetApp FlexCache data.

NetApp FlexCache volumes will be created during the FlexCache creation on destination NetApp ONTAP.

CLI

Create aggregate:

```
storage aggregate create -aggregate <aggr_vultr> -disklist <connected-disk> -node <node>


ONTAPSelectCluster::> aggr create -aggregate aggr_data -disklist NET-1.1 -ha-policy
sfo -node ONTAPSelectCluster-01
Info: The layout for aggregate "aggr_data" on node "ONTAPSelectCluster-01" would be:
       First Plex
        RAID Group rg0, 1 disks (advanced_zoned checksum, raid0)
                                                                 Usable Physical
          Position   Disk                      Type              Size     Size
          ---------- ------------------------- ---------- -------- --------
          data       NET-1.1                   SSD          3.44TB   3.50TB
      Aggregate capacity available for volume use would be 3.10TB.
      3.50TB would be used from capacity license.
 Do you want to continue? {y|n}: y
[Job 40] Job succeeded: DONE
ONTAPSelectCluster::>
```

Validation

```
 aggregate show

ONTAPSelectCluster::> aggr show


Aggregate     Size Available Used% State   #Vols  Nodes            RAID Status
--------- -------- --------- ----- ------- ------ ---------------- ------------
aggr0_ONTAPSelectCluster_01
           60.22GB    2.92GB   95% online       1 ONTAPSelectClust raid0,
                                                  er-01            normal
aggr_data   3.10TB    3.10TB    0% online       0 ONTAPSelectClust raid0,
                                                  er-01            normal
2 entries were displayed.
```


Expected Output

- Aggregate aggr_data created successfully
- Shows associated **node** (ONTAPSelectCluster-01) and **disks** (NET-1.1)
- Aggregate **status** is online and ready to host volumes
- Displays **total size** and **available space** of the aggregate

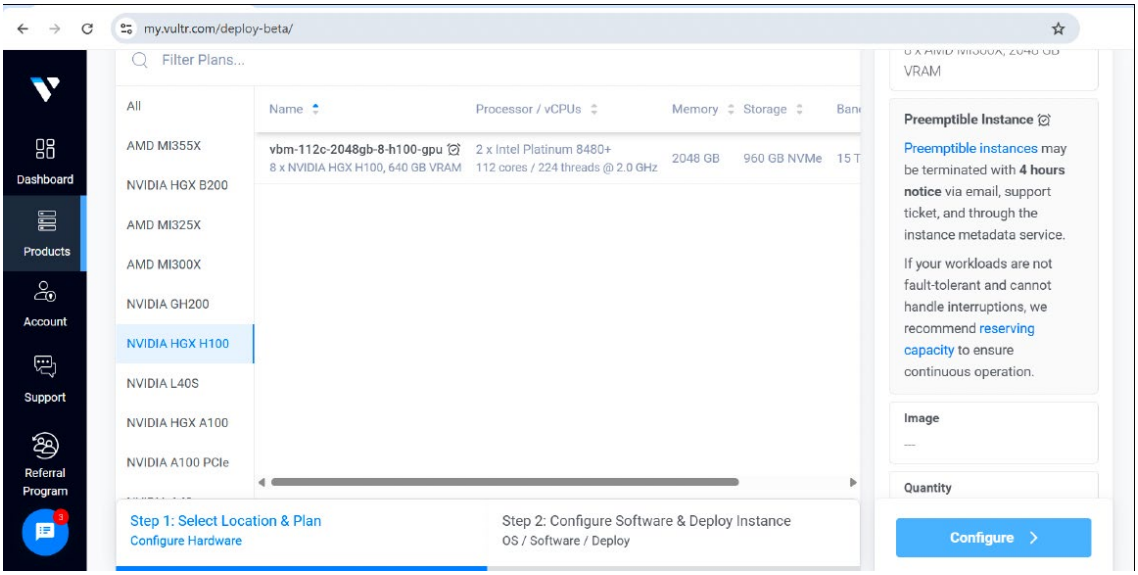
Create a compute node with NVIDIA GPUs

Provision a compute node with NVIDIA GPUs sized appropriately for your application and performance requirements. Deploy the compute node within the same Virtual Private Cloud (VPC) as the NetApp FlexCache destination system to ensure low-latency, local network access and optimal data performance.

Key Steps:

- **Provision the Compute Node** – Deploy a GPU-enabled compute instance aligned with workload performance and capacity needs.
- **Select GPU Configuration** – Choose the appropriate NVIDIA model based on workload type, such as training, inference, or data processing.
- **Verify Network Connectivity** – Ensure connectivity between the compute node and storage system is established, with all required ports open for seamless data access.

The following screenshot illustrates some of the available NVIDIA GPU options:



Obtain the IP addresses of the Vultr compute nodes provisioned with NVIDIA GPUs, as these will be required for the configuration of export policies on the storage system.

```
root@Nvidia-Compute:~# hostname -I && hostname

149.28.218.96 10.12.96.8

nvidia-gpu-compute

root@Nvidia-Compute:~#
```


```
root@Nvidia-Compute:~# hostname -I && hostname
149.28.218.96 10.12.96.8
nvidia-gpu-compute
root@Nvidia-Compute:~#
```

Configure volume export/security settings (NFS/SMB)

Export policies define which compute nodes can mount the FlexCache volume via NFS or SMB. Vultr compute instances must be explicitly granted both read-write access to the destination volumes. We will be configuring the following:

- **export-policy create** – Creates an export policy that defines the overall access framework for NFS clients on the SVM.
- **export-policy rule create** – Adds specific access rules (read-only/read-write, security type, client conditions) to the policy so ONTAP knows what permissions to enforce.
- **volume modify -policy <policy-name>** – Applies the export policy to a specific volume, enabling ONTAP to enforce those access rules when the volume is mounted.
- **volume mount -junction-path <path>** – Mounts the volume into the SVM's namespace, making it visible and accessible for NFS clients to mount.

```
vserver export-policy create -vserver dst_svm -policyname <policy-name>
```

```
ONTAPSelectCluster::> export-policy create -policyname Test_Expolicy -vserver SVM_data
```

Validate:

```
export-policy show
```

```
ONTAPSelectCluster::> export-policy show
Vserver          Policy Name
---------------  -------------------
SVM_data         Test_Expolicy
SVM_data         default
2 entries were displayed.
```

Expected Output

- Export policy Test_Expolicy created successfully
- Associated with the **Vserver** SVM_data
- Status indicates the policy is **active** and ready to be configured with rules
- Confirms policy is ready to control NFS/CIFS access for volumes under the Vserver

Add rule:

After creating the policy, associate the required hosts with the policy and assign the appropriate permission levels.

```
vserver export-policy rule create -vserver dst_svm -policyname <policy-name>-
clientmatch <nfs_node_on_vultr> -rorule any -rwrule any -protocol nfs3 –superuser
any

ONTAPSelectCluster::*> export-policy rule create -policyname Test_Expolicy
-clientmatch 10.12.96.8 -rorule any -rwrule any -vserver SVM_data -protocol nfs3
-superuser any
```

Validate:

```
export-policy rule show -policyname Test_Expolicy -clientmatch 10.12.96.8
```

```
ONTAPSelectCluster::> export-policy rule show -policyname Test_Expolicy -clientmatch 10.12.96.8
            Policy          Rule    Access   Client                RO
Vserver     Name            Index   Protocol Match                 Rule
----------- --------------- ------  -------- --------------------- ---------
SVM_data    Test_Expolicy   4       nfs3     10.12.96.8            any
```

```
export-policy rule show -policyname Test_Expolicy -fields rorule,rwrule

ONTAPSelectCluster::> export-policy rule show -policyname Test_Expolicy -fields rorule,rwrule
vserver  policyname    ruleindex rorule rwrule
-------- ------------- --------- ------ ------
SVM_data Test_Expolicy 3         any    any
```

Expected Output

- Rule added successfully to **export policy** Test_Expolicy
- Shows **client match:** 10.12.96.8
- Read-only (rorule) and read-write (rwrule) permissions set (any)
- Protocol applied: **NFSv3**
- Superuser access: **any**
- Confirms the rule is **active** and ready to allow NFS access for the client

Create Destination NetApp FlexCache Volume

A NetApp FlexCache volume stores hot (frequently accessed) data close to users or applications, while the origin volume remains the authoritative source. FlexCache automatically manages cache coherency, metadata synchronization, and write-forward behavior, ensuring consistent access across sites without administrative overhead.

FlexCache volumes do not need to match the size of the origin volume. Because they store only hot data, metadata, and recently accessed blocks, they can be provisioned significantly smaller - typically 5 - 20% of the origin volume's capacity - making them highly storage-efficient.

CLI

Create a NetApp FlexCache volume

The FlexCache volume is created on the destination cluster/SVM and linked to the source volume. Only metadata and frequently accessed data are cached, reducing latency and WAN traffic.

```
volume flexcache create -vserver <cache_svm> -volume <cache_volume_name> -aggr-list
<aggregate> -size 1TB -origin-volume <origin_vol> -origin-vserver <origin_svm>

ONTAPSelectCluster::> volume flexcache create -volume Flex_Test_DES01 -vserver SVM_
data -aggr-list aggr_data -size 6GB -origin-volume Flex_Test_S10 -origin-vserver sx
-junction-path /Flex_Test_DES01

[Job 95] Job succeeded: Successful.
```

- **volume flexcache create**
  Initiates the creation of a FlexCache volume on the destination (cache) cluster.
- **-vserver <cache_svm>**
  Specifies the Storage Virtual Machine (SVM) where the FlexCache volume will be created.
- **-volume <cache_volume>**
  Defines the name of the FlexCache volume on the cache SVM.
- **-origin-vserver <source_svm>**
  Identifies the source SVM that hosts the original (origin) volume.
- **-origin-volume <source_volume>**
  Specifies the source volume whose data will be cached in the FlexCache volume.
- **-size <size>**
  Sets the size of the FlexCache volume (used for metadata and cached data blocks).

Validate from Destination cluster

Confirms the cache is correctly associated with the origin volume.

```
volume flexcache show

ONTAPSelectCluster::> volume flexcache show
Vserver Volume      Size       Origin-Vserver Origin-Volume Origin-Cluster
------- ----------- ---------- -------------- ------------- --------------
SVM_data Flex_Test_DES01 6GB    sx            Flex_Test_S10 sxId0edb1927eae795ba7
```

Validate from Source origin NetApp ONTAP cluster

Validate origin and cache relationship health.

```
volume flexcache origin show
sxId0edb1927eae795ba7::> volume flexcache origin show
Origin-Vserver Origin-Volume  Cache-Vserver  Cache-Volume  Cache-Cluster
-------------- -------------- -------------- ------------- --------------
sx            Flex_Test_S10  SVM_data       Flex_Test_DES01 ONTAPSelectCluster
```

Expected Output

- FlexCache volume Flex_Test_DES01 created successfully
- Associated with **Vserver** SVM_data
- Linked to **aggregate** aggr_data and sized **6GB**
- Origin volume: Flex_Test_S10 on Vserver sx
- Caching relationship with origin volume established and ready to serve cached data to clients

Mount the NetApp FlexCache Volume with Junction Path

The volume mount command attaches the volume to the SVM's namespace at a junction path, making it visible and mountable by NFS clients. Without a junction path, the volume exists internally but cannot be accessed over NFS.

```
Volume show -volume <volumename> -fields junction-path
```

```
ONTAPSelectCluster::*> vol mount -volume Flex_vol_Test01 -vserver SVM_Dataflex -junction-
path /Flex_vol_Test01
```

```
ONTAPSelectCluster::> vol mount -volume Flex_Test_DES01 -junction-path /Flex_Test_DES01 -vserver SVM_data
```

Validate

```
Volume show -volume <volumename> -fields junction-path
```

```
ONTAPSelectCluster::*> vol show -volume Flex_vol_Test01 -fields junction-path
vserver      volume          junction-path
------------ --------------- ----------------
SVM_Dataflex Flex_vol_Test01 /Flex_vol_Test01
```

Expected Output

- Volume Flex_Test_DES01 mounted successfully
- Mounted on **Vserver** SVM_data
- Junction path set to /Flex_Test_DES01
- Volume status is **online** and accessible to clients
- Confirms the volume is ready for read/write operations

Modify the volume with export-policy

The volume modifies command assigns an export policy to the volume, enabling ONTAP to apply the correct access rules when clients attempt to mount it. Without an export policy bound to the volume, ONTAP will not allow any NFS access.

Validate policy before modifying:

```
ONTAPSelectCluster::> vol show Flex_Test_DES01 -fields policy
vserver  volume          policy
-------- --------------- -------
SVM_data Flex_Test_DES01 default
```

Modify the policy:

```
volume modify -vserver dst_svm –volume <vol> -policy <policy-name>
```

```
ONTAPSelectCluster::> vol modify -volume Flex_Test_DES01 -vserver SVM_data -policy
Test_Expolicy
[Job 104] Job succeeded: volume modify succeeded
```

Validate policy after modifying:

```
volume show –volume <volume> -fields policy
```

```
ONTAPSelectCluster::> vol show -volume Flex_Test_DES01 -vserver SVM_data -fields policy
vserver  volume          policy
-------- --------------- -------------
SVM_data Flex_Test_DES01 Test_Expolicy
```

Expected Output

- Volume Flex_Test_DES01 updated successfully
- Associated with **Vserver** SVM_data
- **Export policy** changed/applied to Test_Expolicy
- Confirms the volume is **online** and the new policy is active
- Volume is ready for client access under the updated export policy

Check Export Policy access

The **export-policy check-access** command in NetApp ONTAP is used to **verify whether a specific client is allowed to access an NFS export and what level of access it will receive.** It simulates an NFS access request and evaluates the export policy rules without mounting the volume. The command checks the client IP or hostname against the export policy rules, determines whether access is permitted or denied, and reports the effective permissions such as read-only, read-write, superuser access, and protocol version. This is mainly used for **troubleshooting NFS permission issues,** helping administrators quickly confirm if export policies are correctly configured for a given client.

```
export-policy check-access -vserver SVM_data -volume Flex_Test_DES01 -client-ip
10.12.96.8 -authentication-method sys -protocol nfs3 -access-type read-write
```

If access is denied, add clientmatch into the root volume export policy as below:

```
ONTAPSelectCluster::*> export-policy rule create -policyname default -vserver SVM_
Data -clientmatch 10.12.96.8 -rorule any -rwrule any -superuser any
```

Validate:

```
ONTAPSelectCluster::> export-policy check-access -vserver SVM_data -volume Flex_
Test_DES01 -client-ip 10.12.96.8 -authentication-method sys -protocol nfs3 -access-
type read-write
                                  Policy    Policy     Rule            Security
Path                 Policy       Owner     Owner Type Index Access    Style
-------------------- ----------   --------- ---------- ----- ---------- ----------
/                    default      SVM_data_root volume     3  read       unix
//Test_Vol           Test_Expolicy Test_Vol volume          4  read-write mixed
2 entries were displayed.
```

Expected Output

- Checks access for **client IP** 10.12.96.8 on volume Flex_Test_DES01
- Shows **Vserver:** SVM_data
- Authentication method: sys
- Protocol: NFSv3
- Access type tested: read-write
- Output indicates whether access is **allowed** or **denied**
- Confirms which **export policy rule** permits or blocks the access

Test Mount on Vultr Compute

Verification that Vultr compute nodes can access the replicated dataset over NFS. Ensures the end-to-end path - from **on-prem** » **NetApp FlexCache** » **NetApp ONTAP(Destination)** » **Vultr compute** - is fully operational.

When we mount an NFS export from NetApp ONTAP(Destination) in Vultr:

- We are mounting the **active filesystem** of the destination NetApp FlexCache Volume
- NetApp FlexCache maintains a local cache of hot (frequently accessed) data from the source volume. Clients can perform read and write operations on the NetApp FlexCache volume - reads are served locally from the cache, while all writes are sent directly to the origin (source) volume, which remains the authoritative data source.

CLI (from compute instance on Vultr Cloud)

```
mount -t nfs <ontap_select_ip>:/<junction-path> <compute node mount point>
```

```
root@nvidia-gpu-compute:~# mount -t nfs 10.12.96.52:/Flex_Test_DES01 /mnt/Flexcache_Test
root@nvidia-gpu-compute:~#
```

```
root@nvidia-gpu-compute:~# mount -t nfs 10.12.96.52:/Flex_Test_DES01 /mnt/
Flexcache_Test
```

Testing Read Access

Once the FlexCache volume is mounted, we can verify that the end user is able to successfully access and read the cached data from the cache volume.

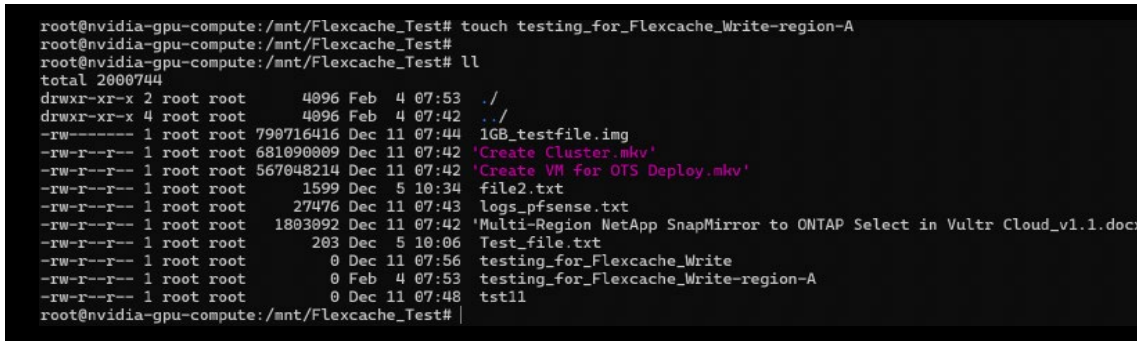
```
cat /mnt/Flexcache_Test/<file>
```

```
root@nvidia-gpu-compute:/mnt/Flexcache_Test# cat logs_pfsense.txt
Last 500 IPsec Log Entries. (Maximum 500)
Nov 28 09:08:55 14[CFG]          remote_port = 500
Nov 28 09:08:55 14[CFG]          send_certreq = 1
Nov 28 09:08:55 14[CFG]          send_cert = CERT_SEND_IF_ASKED
Nov 28 09:08:55 14[CFG]          ppk_id = (null)
Nov 28 09:08:55 14[CFG]          ppk_required = 0
Nov 28 09:08:55 14[CFG]          mobike = 0
Nov 28 09:08:55 14[CFG]          aggressive = 0
Nov 28 09:08:55 14[CFG]          dscp = 0x00
Nov 28 09:08:55 14[CFG]          encap = 0
```

```
root@nvidia-gpucompute-01:/mnt/Flexcache_Test# cat logs_pfsense.txt
Last 500 IPsec  Log Entries. (Maximum 500)
Nov 28 09:08:55 14[CFG]          remote_port = 500
Nov 28 09:08:55 14[CFG]          send_certreq = 1
Nov 28 09:08:55 14[CFG]          send_cert = CERT_SEND_IF_ASKED
Nov 28 09:08:55 14[CFG]          ppk_id = (null)
Nov 28 09:08:55 14[CFG]          ppk_required = 0
Nov 28 09:08:55 14[CFG]          mobike = 0
Nov 28 09:08:55 14[CFG]          aggressive = 0
Nov 28 09:08:55 14[CFG]          dscp = 0x00
Nov 28 09:08:55 14[CFG]          encap = 0
```

Testing Write Access (Destination NetApp ONTAP)

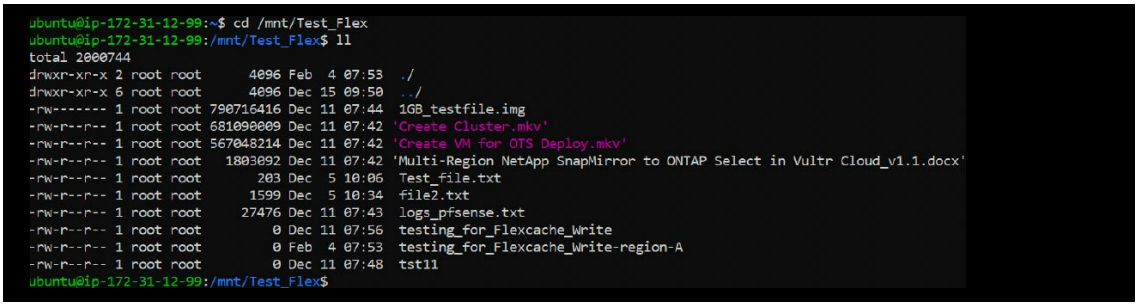
The following commands demonstrate the testing of FlexCache write functionality. A file named Testing_for_FlexCache_Write was successfully created from the cache volume, and the same file is now available and visible on the source NFS host, confirming that write operations are correctly redirected to the source volume.




```
root@nvidia-gpu-compute:/mnt/Flexcache_Test# touch testing_for_Flexcache_Write-
region-A
root@nvidia-gpu-compute:/mnt/Flexcache_Test#
root@nvidia-gpu-compute:/mnt/Flexcache_Test# ll
total 2000744
drwxr-xr-x 2 root root      4096 Feb 4 07:53  ./
drwxr-xr-x 4 root root      4096 Feb 4 07:42  ../
-rw------- 1 root root 790716416 Dec 11 07:44 1GB_testfile.img
-rw-r--r-- 1 root root 681090009 Dec 11 07:42 'Create Cluster.mkv'
-rw-r--r-- 1 root root 567048214 Dec 11 07:42 'Create VM for OTS Deploy.mkv'
-rw-r--r-- 1 root root      1599 Dec 5 10:34   file2.txt
-rw-r--r-- 1 root root     27476 Dec 11 07:43  logs_pfsense.txt
-rw-r--r-- 1 root root   1803092 Dec 11 07:42 'Multi-Region NetApp SnapMirror to
ONTAP Select in Vultr Cloud_v1.1.docx'
-rw-r--r-- 1 root root       203 Dec 5 10:06   Test_file.txt
-rw-r--r-- 1 root root         0 Dec 11 07:56  testing_for_Flexcache_Write
-rw-r--r-- 1 root root         0 Feb 4 07:53   testing_for_Flexcache_Write-region-A
-rw-r--r-- 1 root root         0 Dec 11 07:48  tst11
root@nvidia-gpu-compute:/mnt/Flexcache_Test#
```


Testing Write Through to Origin (Source NetApp ONTAP)

ONTAP FlexCache uses **write-through semantics**, where all writes are forwarded directly to the origin volume, preserving a single authoritative source of truth. A write or update issued at the **FlexCache mount** is **forwarded to the origin volume.**



```
ubuntu@ip-172-31-12-99:~$ cd /mnt/Test_Flex
ubuntu@ip-172-31-12-99:/mnt/Test_Flex$ ll
total 2000744
drwxr-xr-x 2 root root      4096 Feb 4 07:53 ./
drwxr-xr-x 6 root root      4096 Dec 15 09:50 ../
-rw------- 1 root root 790716416 Dec 11 07:44 1GB_testfile.img
-rw-r--r-- 1 root root 681090009 Dec 11 07:42 'Create Cluster.mkv'
-rw-r--r-- 1 root root 567048214 Dec 11 07:42 'Create VM for OTS Deploy.mkv'
-rw-r--r-- 1 root root   1803092 Dec 11 07:42 'Multi-Region NetApp SnapMirror to
ONTAP Select in Vultr Cloud_v1.1.docx'
-rw-r--r-- 1 root root       203 Dec 5 10:06   Test_file.txt
-rw-r--r-- 1 root root      1599 Dec 5 10:34   file2.txt
-rw-r--r-- 1 root root     27476 Dec 11 07:43  logs_pfsense.txt
-rw-r--r-- 1 root root         0 Dec 11 07:56  testing_for_Flexcache_Write
-rw-r--r-- 1 root root         0 Feb 4 07:53   testing_for_Flexcache_Write-region-A
-rw-r--r-- 1 root root         0 Dec 11 07:48  tst11
ubuntu@ip-172-31-12-99:/mnt/Test_Flex$
```

Expected Output

- Mount is successful
- Read and Write operations success

Validate Throughput and Performance

A performance test using read workloads to ensure ONTAP meets expected throughput for AI/analytics. Confirms the system can sustain the I/O patterns required by downstream GPU/compute pipelines.

The dd test is run twice to demonstrate FlexCache behavior: the first run reads data from the origin across the WAN, while the second run reads the same data locally from the FlexCache, delivering higher throughput.

CLI (compute-side example)

```
dd if=/mnt/Flexcache_Test/<bigfile> of=/dev/null bs=1M status=progress
```

```
root@nvidia-gpu-compute:/mnt/Flexcache_Test# dd if=cluster.mkv of=/dev/null bs=1M status=progress
673185792 bytes (673 MB, 642 MiB) copied, 20 s, 33.5 MB/s
649+1 records in
649+1 records out
681090009 bytes (681 MB, 650 MiB) copied, 20.3648 s, 33.4 MB/s
root@nvidia-gpu-compute:/mnt/Flexcache_Test#
```

Expected Output (from across the WAN)

- Stable read throughput (e.g., **20–200** MB/s depending on network access to the source over the WAN)
- No read stalls or NFS timeout messages

Now clear linux cache (so test doesn't take data from memory but from FlexCache)

```
echo 3 > /proc/sys/vm/drop_caches
```

```
root@nvidia-gpu-compute:/mnt/Flexcache_Test# echo 3 > /proc/sys/vm/drop_caches
root@nvidia-gpu-compute:/mnt/Flexcache_Test#
root@nvidia-gpu-compute:/mnt/Flexcache_Test#
root@nvidia-gpu-compute:/mnt/Flexcache_Test# dd if=cluster.mkv of=/dev/null bs=1M status=progress
666894336 bytes (667 MB, 636 MiB) copied, 2 s, 333 MB/s
649+1 records in
649+1 records out
681090009 bytes (681 MB, 650 MiB) copied, 2.04459 s, 333 MB/s
root@nvidia-gpu-compute:/mnt/Flexcache_Test# ^C
```

Expected Output (from FlexCache)

- Stable read throughput (e.g., **200–400** MB/s)
- No read stalls or NFS timeout messages

Observation: Higher Performance Gains with FlexCache

Accessing data through ONTAP FlexCache can deliver significantly higher throughput than reading directly from the origin volume over the WAN, because frequently accessed (hot) data is served from the FlexCache volume that is mounted within the same region or site as the client. Although FlexCache is accessed over the network, it avoids WAN round trips by servicing read requests from a nearby cache, reducing latency and offloading read traffic from the origin volume. This enables near–local-storage performance for repeated read operations while preserving a single authoritative source of truth at the origin.

Step 4 – Configure Second Region (Region B) for FlexCache on Vultr Cloud

Configure ONTAP Cluster Peering

This section assumes that **Region B** is already configured as part of the same VPN and that **cluster peering has been successfully established**, like the setup completed for **Region A**. The following steps describe the procedure to **mount a volume from Region B**.

This section covers:

- Configure cluster peering (bidirectional trust)
- Configure SVM peering (data SVM to data SVM authorization)
- Network Test-Path (NetApp FlexCache connectivity validation)
- Prepare destination aggregate
- Configure volume export/security settings (NFS/SMB)
- Create destination NetApp FlexCache Volume
- Test mounts on Vultr Compute

Cluster Peering

Cluster peering for Region B is configured in the same manner as Region A. Refer to Cluster Peering.

Run the '`cluster peer create`' command first on the destination side - the side that will RECEIVE the peer request.

On Netapp ONTAP on Vultr Cloud

```
network interface show -role intercluster
```

```
ONTAPSelectCluster::> network interface show -role intercluster
            Logical    Status     Network            Current       Current Is
Vserver     Interface  Admin/Oper Address/Mask       Node          Port    Home
----------- ---------- ---------- ------------------ ------------- ------- ----
ONTAPSelectCluster
            ic1          up/up    10.12.96.51/20     ONTAPSelectCluster-01 e0a true
```

On On-Prem ONTAP

```
sxId0edb1927eae795ba7::> network interface show -role intercluster
             Logical    Status            Network          Current    Current Is
 Vserver     Interface  Admin/Oper        Address/Mask     Node       Port    Home
 ----------- ---------- ---------------- ------------------ ---------- ------- ----
 sxId0edb1927eae795ba7
                inter_1       up/up     172.31.157.201/20
sxId0edb1927eae795ba7-01 e0e   true
                inter_2       up/up     172.31.144.104/20
sxId0edb1927eae795ba7-02 e0e   true
 2 entries were displayed.
```

Expected Output

- status-admin = up, status-oper = up
- Correct IPs assigned
- Home-node and home-port match the configuration

CLI for cluster peering to be run on Vultr Cloud NetApp ONTAP:

```
cluster peer create -peer-addrs <peer-intercluster-LIF-IP of ON-PREM> -generate-
passphrase true
```

```
ONTAPSelectCluster::*> cluster peer show
This table is currently empty.

ONTAPSelectCluster::*> cluster peer create -peer-addrs 172.31.144.104 -generate-
passphrase true

Notice:
 Passphrase: tl++CrpFI1Y/Axxp+3YzlU+R
 Expiration Time: 12/3/2025 06:37:24 +00:00
 Initial Allowed Vserver Peers: -
 Intercluster LIF IP: 10.12.96.51
 Peer Cluster Name: sxId0edb1927eae795ba7

 Warning: make a note of the passphrase - it cannot be displayed again.
```

Note: Copy the Passphrase which will be needed in the next command

CLI for cluster peering to be run on On-Prem NetApp ONTAP:

Here, don't pass '`-generate-passphrase true`', as we need to use the generated passphrase from NetApp ONTAP[Destination]

- **Passphrase is generated only once** - on the destination cluster that initiates the peering (Destination NetApp ONTAP).
- The **on-prem source** must **reuse** that same passphrase when replying to complete the peering.

```
cluster peer create -peer-addrs <peer-intercluster-LIF-IP of ON-PREM>
```

```
sxId0edb1927eae795ba7::> cluster peer create -address-family ipv4 -peer-addrs
10.12.96.51

Notice: Use a generated passphrase or choose a passphrase of 8 or more characters.
To ensure the authenticity of the peering relationship, use a phrase or sequence of
characters that would be hard to guess.

Enter the passphrase:
Confirm the passphrase:
```

Validation

```
cluster peer show
```

```
ONTAPSelectCluster::*> cluster peer show
Peer Cluster Name         Cluster Serial Number Availability   Authentication
------------------------- --------------------- -------------- --------------
sxId0edb1927eae795ba7     1-80-000011           Available      ok
```



```
sxId0edb1927eae795ba7::> cluster peer show
Peer Cluster Name         Cluster Serial Number Availability   Authentication
------------------------- --------------------- -------------- --------------
ONTAPSelectCluster        1-80-000011           Available      ok
```

Expected Output

- Availability = **Available**
- Authentication = **ok**
- Remote cluster name displayed
- No timeout or “unreachable” errors

SVM Peering

Since the Vultr Cloud ONTAP system will act as the FlexCache destination, it is the appropriate place to initiate the ‘`vserver peer create’` command. The on-premises ONTAP system will then receive and accept the peering request.

Before creating the vserver peering, ensure the required vserver and data LIFs are created and configured to support client access and peering operations.

Create a vserver:

```
ONTAPSelectCluster::>  vserver create -vserver SVM_Dataflex -aggregate aggr_data -subtype default -rootvolume SVM_Dataflex_root -root
volume-security-style mixed -language C.UTF-8 -snapshot-policy default -data-services data-iscsi,data-nfs,data-cifs,data-flexcache,da
ta-nvme-tcp
[Job 123] Job succeeded:
Vserver creation completed.
```

Validate:

```
ONTAPSelectCluster::> vserver show
                                 Admin      Operational Root
Vserver     Type    Subtype      State      State       Volume     Aggregate
----------- ------- ---------- ---------- ----------- ---------- ----------
ONTAPSelectCluster admin -      -          -           -          -
ONTAPSelectCluster-01 node -    -          -           -          -
SVM_Dataflex data   default    running    running     SVM_       aggr_data
                                                      Dataflex_
                                                      root
SVM_data    data    default    running    running     SVM_data_  aggr_data
                                                      root
```

Data LIF creation:

```
ONTAPSelectCluster::> network interface create -vserver SVM_Dataflex -lif SVM_Dataflex_data_01 -data-protocol fcache,nfs,cifs -addres
s 10.12.96.53 -netmask 255.255.240.0 -home-node ONTAPSelectCluster-01 -home-port e0b
```

Validate:

```
ONTAPSelectCluster::> net int show SVM_Dataflex_data_01
  (network interface show)
            Logical    Status     Network            Current       Current Is
Vserver     Interface  Admin/Oper Address/Mask       Node          Port    Home
----------- ---------- ---------- ------------------ ------------- ------- ----
SVM_Dataflex
            SVM_Dataflex_data_01 up/up 10.12.96.53/20 ONTAPSelectCluster-01 e0b true
```

CLI run on NetApp ONTAP (Vultr Cloud - Destination)

Run the following command on the ONTAP cluster (the destination). This command creates the pending peering relationship.

```
ONTAPSelectCluster::*> vserver peer create -vserver SVM_Dataflex -peer-vserver SVM_
Flexdata -applications snapmirror,flexcache -peer-cluster sxId0edb1927eae795ba7

Info: [Job 125] 'vserver peer create' job queued
```

CLI run on On-prem NetApp ONTAP [Source]

Immediately after the create command, run the following command on the corresponding **On-Prem cluster** (the source) to accept the peering request and finalize the relationship.

```
sxId0edb1927eae795ba7::*> vserver peer accept -vserver SVM_Flexdata -peer-vserver
SVM_Dataflex

Info: [Job 245] 'vserver peer accept' job queued
```

Validation from both ONTAP (Destination) and On-prem ONTAP (Source)

```
vserver peer show
```

Validation:

CLI (Source)

```
ONTAPSelectCluster::*> vserver peer show
            Peer        Peer                           Peering        Remote
Vserver     Vserver     State        Peer Cluster      Applications   Vserver
----------- ----------- ------------ ----------------- -------------- ---------
SVM_Dataflex SVM_Flexdata peered      sxId0edb1927eae795ba7 snapmirror, flexcache
SVM_Flexdata
SVM_data    sx           peered      sxId0edb1927eae795ba7 flexcache, snapmirror sx
2 entries were displayed.
```

CLI (Destination)

```
sxId0edb1927eae795ba7::> vserver peer show
            Peer        Peer                           Peering        Remote
Vserver     Vserver     State        Peer Cluster      Applications   Vserver
----------- ----------- ------------ ----------------- -------------- ---------
SVM_Flexdata SVM_Dataflex peered      ONTAPSelectCluster snapmirror, flexcache
SVM_Dataflex
sx           SVM_data    peered      ONTAPSelectCluster flexcache, snapmirror
SVM_data
2 entries were displayed.
```

Expected Output

- Peer state = **peered**
- Applications = **FlexCache, snapmirror**
- No "initial" or "pending" states
- No peer conflicts

Network Test-Path

The network test-path command is a specialized **NetApp ONTAP utility** used to **proactively validate the entire network path** necessary for NetApp FlexCache and cluster peering operations. It is the most robust way to verify your network is ready.

This command should be run **after** Cluster Peering and vserver peering is set up but **before** you attempt to set up NetApp FlexCache.

This tests the following:

- **Connectivity & Routing:** Confirms that a dedicated SnapMirror and NetApp FlexCache connection can be established between the source and destination cluster nodes over the correct **Intercluster LIFs.**
- Firewall Status: Verifies that the firewall allows traffic for both the **SnapMirror and NetApp FlexCache control channel (TCP 11104)** and the **data transfer channel (TCP 10000)**.
- **LIF Configuration:** Ensures the Intercluster LIFs on both the source and destination are properly configured, up, and listening for SnapMirror and NetApp FlexCache traffic.

Run this from both ONTAP and On-prem ONTAP:

```
network test-path -source-node <Source_Node_Name> -destination-cluster
<Destination_Cluster_Name> -destination-node <Destination_Node_Name>
```

```
sxId0edb1927eae795ba7::*> network test-path -source-node sxId0edb1927eae795ba7-01
-destination-cluster ONTAPSelectCluster -destination-node ONTAPSelectCluster-01

Warning: This operation will generate large amount of cluster traffic and can cause
temporary cluster traffic slowness.

Do you want to continue? {y|n}: y

Initiating path test. It can take up to 120 seconds for results to be displayed.
 Test Duration: 14.25 secs
 Send Throughput: 27.97 MB/sec
Receive Throughput: 27.97 MB/sec
 MB Sent: 398.62
 MB Received: 398.62
 Avg Latency: 5157.99 ms
```

```
sxId0edb1927eae795ba7::*> network test-path -source-node sxId0edb1927eae795ba7-02
-destination-cluster ONTAPSelectCluster -destination-node ONTAPSelectCluster-01
Warning: This operation will generate large amount of cluster traffic and can cause
temporary cluster traffic slowness.
Do you want to continue? {y|n}: y

 Test Duration: 14.26 secs
 Send Throughput: 29.64 MB/sec
Receive Throughput: 29.64 MB/sec
 MB Sent: 422.56
 MB Received: 422.56
 Avg Latency: 4797.68 ms
```

```
ONTAPSelectCluster::*> network test-path -source-node ONTAPSelectCluster-01
-destination-cluster sxId0edb1927eae795ba7 -destination-node
sxId0edb1927eae795ba7-01

Warning: This operation will generate large amount of cluster traffic and can cause
temporary cluster traffic slowness.
Do you want to continue? {y|n}: y

Initiating path test. It can take up to 120 seconds for results to be displayed.
 Test Duration: 14.23 secs
 Send Throughput: 20.71 MB/sec
Receive Throughput: 20.71 MB/sec
 MB Sent: 294.69
 MB Received: 294.69
 Avg Latency: 5387.71 ms
```

Expected Output

- Confirms network connectivity between source and destination nodes
- Shows **source node** and **destination node/cluster**
- Displays **status** of the path (e.g., success or failure)
- Provides **latency** or **response** time details (if applicable)
- Indicates any issues such as packet loss or unreachable network

Prepare Destination Aggregate

Aggregates are storage pools, and FlexVols are the ONTAP volumes that will receive replicated NetApp FlexCache data.

NetApp FlexCache volumes will be created during the FlexCache creation on destination NetApp ONTAP.

CLI

Create aggregate:

```
storage aggregate create -aggregate <aggr_vultr> -disklist <connected-disk> -node
<node>
```



```
ONTAPSelectCluster::> aggr create -aggregate aggr_data -disklist NET-1.1 -ha-policy
sfo -node ONTAPSelectCluster-01

Info: The layout for aggregate "aggr_data" on node "ONTAPSelectCluster-01" would
be:

      First Plex

          RAID Group rg0, 1 disks (advanced_zoned checksum, raid0)
                                                            Usable   Physical
            Position   Disk                     Type        Size     Size
            ---------- ------------------------ ---------- -------- --------
            data       NET-1.1                   SSD        3.44TB   3.50TB

 Aggregate capacity available for volume use would be 3.10TB.
 3.50TB would be used from capacity license.

Do you want to continue? {y|n}: y
[Job 40] Job succeeded: DONE

ONTAPSelectCluster::>
```

Validation

```
aggregate show
```

```
ONTAPSelectCluster::> aggr show

Aggregate     Size Available Used% State   #Vols  Nodes            RAID Status
--------- -------- --------- ----- ------- ------ ---------------- ------------
aggr0_ONTAPSelectCluster_01
          60.22GB    2.92GB   95% online       1 ONTAPSelectClust raid0,
                                                  er-01            normal
aggr_data   3.10TB    3.10TB    0% online       0 ONTAPSelectClust raid0,
                                                  er-01            normal
2 entries were displayed.
```

Expected Output

- Aggregate aggr_data created successfully
- Shows associated **node** (ONTAPSelectCluster-01) and **disks** (NET-1.1)
- Aggregate **status** is online and ready to host volumes
- Displays **total size** and **available space** of the aggregate

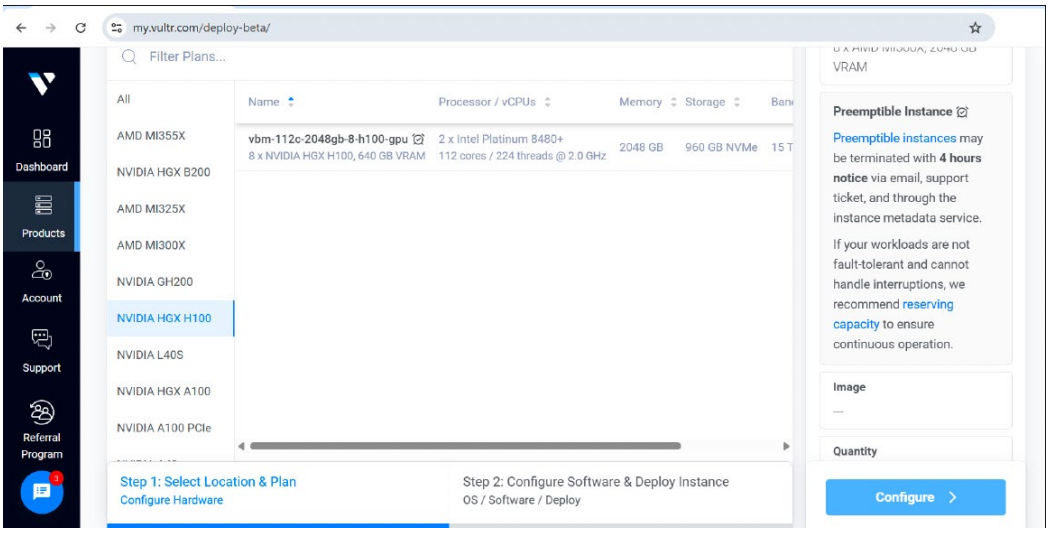
Create a compute node with NVIDIA GPUs

Provision a compute node with NVIDIA GPUs sized appropriately for your application and performance requirements. Deploy the compute node within the same Virtual Private Cloud (VPC) as the NetApp FlexCache destination system to ensure low-latency, local network access and optimal data performance.

Key Steps:

- **Provision the Compute Node** – Deploy a GPU-enabled compute instance aligned with workload performance and capacity needs.
- **Select GPU Configuration** – Choose the appropriate NVIDIA model based on workload type, such as training, inference, or data processing.
- **Verify Network Connectivity** – Ensure connectivity between the compute node and storage system is established, with all required ports open for seamless data access.

The following screenshot illustrates some of the available NVIDIA GPU options:



Obtain the IP addresses of the Vultr compute nodes provisioned with NVIDIA GPUs, as these will be required for the configuration of export policies on the storage system.

```
root@Nvidia-Compute:~# hostname -I && hostname
149.28.218.96 10.12.96.8
nvidia-gpu-compute
root@Nvidia-Compute:~#
```


```
root@Nvidia-Compute:~# hostname -I && hostname
149.28.218.96 10.12.96.8
nvidia-gpu-compute
root@Nvidia-Compute:~#
```


Configure volume export/security settings (NFS/SMB)

Export policies define which compute nodes can mount the FlexCache volume via NFS or SMB. Vultr compute instances must be explicitly granted read-write access to the destination volumes. We will be configuring the following:

- **export-policy create –** Creates an export policy that defines the overall access framework for NFS clients on the SVM.
- **export-policy rule create** – Adds specific access rules (read-only/read-write, security type, client conditions) to the policy so ONTAP knows what permissions to enforce.
- **volume modify -policy <policy-name>** – Applies the export policy to a specific volume, enabling ONTAP to enforce those access rules when the volume is mounted.
- **volume mount -junction-path <path>** – Mounts the volume into the SVM's namespace, making it visible and accessible for NFS clients to mount.

```
vserver export-policy create -vserver dst_svm -policyname <policy-name>
```

```
ONTAPSelectCluster::*> export-policy create -policyname Test_dest_flexpolicy
-vserver SVM_Dataflex
```

Validate:

```
Export-policy show –policyname <policy-name>
```

```
ONTAPSelectCluster::*> export-policy show -policyname Test_dest_flexpolicy
Vserver          Policy Name
---------------  -------------------
SVM_Dataflex     Test_dest_flexpolicy
```

Expected Output

- Export policy **Test_dest_flexpolicy** created successfully
- Associated with the **Vserver SVM_Dataflex**
- Status indicates the policy is **active** and ready to be configured with rules
- Confirms policy is ready to control NFS/CIFS access for volumes under the Vserver

Add rule:

After creating the policy, associate the required hosts with the policy and assign the appropriate permission levels.

```
ONTAPSelectCluster::*> vserver export-policy rule create -policyname Test_dest_
flexpolicy -vserver SVM_Dataflex -clientmatch 10.12.96.8 -rorule any -rwrule any
-superuser any
```

Validate:

```
export-policy rule show -policyname <policy-name>
```

```
ONTAPSelectCluster::*> export-policy rule show -policyname Test_dest_flexpolicy
             Policy               Rule   Access    Client     RO
Vserver      Name                 Index  Protocol  Match      Rule
------------ -----------------    ------ --------  ---------  ---------
SVM_Dataflex  Test_dest_flexpolicy  1      any       10.12.96.8 any
```

Expected Output

- Rule added successfully to **export policy Test_dest_flexpolicy**
- Shows **client match**: 10.12.96.8
- Read-only **(rorule)** and read-write (rwrule) permissions set **(any)**
- Protocol applied: **NFSv3**
- Superuser access: **any**
- Confirms the rule is **active** and ready to allow NFS access for the client

Create Destination NetApp FlexCache Volume

A NetApp FlexCache volume stores hot (frequently accessed) data close to users or applications, while the origin volume remains the authoritative source. FlexCache automatically manages cache coherency, metadata synchronization, and write-forward behavior, ensuring consistent access across sites without administrative

overhead.

FlexCache volumes do not need to match the size of the origin volume. Because they store only hot data, metadata, and recently accessed blocks, they can be provisioned significantly smaller - typically 5 - 20% of the origin volume's capacity - making them highly storage-efficient.

CLI

Create a NetApp FlexCache volume

The FlexCache volume is created on the destination cluster/SVM and linked to the source volume. Only metadata and frequently accessed data are cached, reducing latency and WAN traffic.

```
volume flexcache create -vserver <cache_svm> -volume <cache_volume_name> -aggr-list
<aggregate> -size 1TB -origin-volume <origin_vol> -origin-vserver <origin_svm>
```

```
ONTAPSelectCluster::*> flexcache create -vserver SVM_Dataflex -volume Flex_vol_Test01
-size 7GB -aggr-list aggr_data -origin-volume Flex_Vol_S20 -origin-vserver SVM_
Flexdata

 (volume flexcache create)

[Job 128] Job succeeded: Successful.
```

- **volume flexcache create**
  Initiates the creation of a FlexCache volume on the destination (cache) cluster.
- **-vserver <cache_svm>**
  Specifies the Storage Virtual Machine (SVM) where the FlexCache volume will be created.
- **-volume <cache_volume>**
  Defines the name of the FlexCache volume on the cache SVM.
- **-origin-vserver <source_svm>**
  Identifies the source SVM that hosts the original (origin) volume.
- **-origin-volume <source_volume>**
  Specifies the source volume whose data will be cached in the FlexCache volume.
- **-size <size>**
  Sets the size of the FlexCache volume (used for metadata and cached data blocks).

Validate from Destination cluster

Confirms the cache is correctly associated with the origin volume.

```
volume flexcache show
```



```
ONTAPSelectCluster::*> flexcache show

 (volume flexcache show)

Vserver   Volume            Size       Origin-Vserver Origin-Volume Origin-Cluster
-------   ------            ---------- -------------- ------------- ------------------
SVM_Dataflex Flex_vol_Test01 7GB        SVM_Flexdata   Flex_Vol_S20  sxId0edb1927eae795ba7
SVM_data Flex_Test_DES01    6GB        sx             Flex_Test_S10 sxId0edb1927eae795ba7

2 entries were displayed.
```

Validate from Source origin NetApp ONTAP cluster

Validate origin and cache relationship health.

```
volume flexcache origin show
```

```
sxId0edb1927eae795ba7::*> volume flexcache origin show
Origin-Vserver Origin-Volume  Cache-Vserver  Cache-Volume    Cache-Cluster
-------------- -------------- -------------- -------------   --------------
SVM_Flexdata   Flex_Vol_S20   SVM_Dataflex   Flex_vol_Test01 ONTAPSelectCluster
sx             Flex_Test_S10  SVM_data       Flex_Test_DES01 ONTAPSelectCluster

2 entries were displayed.
```

Expected Output

- FlexCache volume Flex_vol_Test01 created successfully
- Associated with **Vserver SVM_dataflex**
- Linked to aggregate aggr_data and sized **7GB**
- Origin volume: **Flex_vol_Test01** on **Vserver SVM_Flexdata**
- Junction path set: **/Flex_vol_Test01**
- Volume status is **online** and caching relationship with origin volume established and ready to serve cached data to clients

Mount the NetApp FlexCache Volume with Junction Path

The volume mount command attaches the volume to the SVM's namespace at a junction path, making it visible and mountable by NFS clients. Without a junction path, the volume exists internally but cannot be accessed over NFS.

```
volume mount -vserver dest_svm -volume cache_vol1 -junction-path /Junction-path
```

```
ONTAPSelectCluster::*> vol mount -volume Flex_vol_Test01 -vserver SVM_Dataflex
-junction-path /Flex_vol_Test01
```

```
ONTAPSelectCluster::*> vol mount -volume Flex_vol_Test01 -vserver SVM_Dataflex -junction-path /Flex_vol_Test01
```

Validate:

```
Volume show -volume volumename -fields junction-path
```

```
ONTAPSelectCluster::*> vol show -volume Flex_vol_Test01 -fields junction-path
vserver volume junction-path
------------ --------------- -----------------
SVM_Dataflex Flex_vol_Test01 /Flex_vol_Test01
```

Expected Output

- Volume **Flex_vol_Test01** mounted successfully
- Mounted on Vserver SVM_dataflex
- Junction path set to **/Flex_vol_Test01**
- Volume status is **online** and accessible to clients
- Confirms the volume is ready for read/write operations

Modify the volume with export-policy

The volume modifies command assigns an export policy to the volume, enabling ONTAP to apply the correct access rules when clients attempt to mount it. Without an export policy bound to the volume, ONTAP will not allow any NFS access.

Validate policy before modifying

```
ONTAPSelectCluster::*> vol show -volume Flex_vol_Test01 -fields policy
vserver      volume          policy
------------ --------------- -------
SVM_Dataflex  Flex_vol_Test01  default
```

Modify the volume:

```
volume modify -vserver dst_svm -volume <vol> -policy <policy-name>
```

```
ONTAPSelectCluster::*> vol modify -volume Flex_vol_Test01 -policy Test_dest_
flexpolicy -vserver SVM_Dataflex

[Job 133] Job succeeded: volume modify succeeded
```

```
ONTAPSelectCluster::*> vol show -volume Flex_vol_Test01 -fields policy
vserver      volume          policy
------------ --------------- -------
SVM_Dataflex Flex_vol_Test01 default

ONTAPSelectCluster::*> vol modify -volume Flex_vol_Test01 -policy Test_dest_flexpolicy -vserver SVM_Dataflex
[Job 133] Job succeeded: volume modify succeeded
```

Validation after modifying:

```
volume show –volume <volume> -fields policy
```

```
ONTAPSelectCluster::*> vol show -volume Flex_vol_Test01 -vserver SVM_Dataflex -fields
policy
vserver      volume          policy
------------ --------------- --------------------
SVM_Dataflex  Flex_vol_Test01 Test_dest_flexpolicy
```

```
ONTAPSelectCluster::*> vol show -volume Flex_vol_Test01 -vserver SVM_Dataflex -fields policy
vserver      volume          policy
------------ --------------- --------------------
SVM_Dataflex Flex_vol_Test01 Test_dest_flexpolicy
```

Expected Output

- Volume **Flex_vol_Test01** updated successfully
- Associated with Vserver **SVM_dataflex**
- Export policy changed/applied to **Test_dest_flexpolicy**
- Confirms the volume is **online** and the new policy is active
- Volume is ready for client access under the updated export policy

Check Export Policy access:

The **export-policy check-access** command in NetApp ONTAP is used to **verify whether a specific client is allowed to access an NFS export and what level of access it will receive.** It simulates an NFS access request and evaluates the export policy rules without actually mounting the volume. The command checks the client IP or hostname against the export policy rules, determines whether access is permitted or denied, and reports the effective permissions such as read-only, read-write, superuser access, and protocol version. This is mainly used for **troubleshooting NFS permission issues,** helping administrators quickly confirm if export policies are correctly configured for a given client.

```
export-policy check-access -vserver SVM_Dataflex -volume Flex_vol_Test01 -client-ip
10.12.96.8 -authentication-method sys -protocol nfs3 -access-type read-write
```

If access is denied, add clientmatch into the root volume export policy as below:

```
ONTAPSelectCluster::*> export-policy rule create -policyname default -vserver SVM_
Dataflex -clientmatch 10.12.96.8 -rorule any -rwrule any -superuser any
```

Validate:

```
ONTAPSelectCluster::*> export-policy check-access -vserver SVM_Dataflex -volume Flex_vol_
Test01 -client-ip 10.12.96.8 -authentication-method sys -protocol nfs3 -access-type read-
write

                                  Policy           Policy     Rule             Security
Path                 Policy       Owner            Owner Type Index  Access     Style
-------------------- ---------- ---------------- ---------- -----  ---------- ----------
/                    default    SVM_Dataflex_root volume         2  read       mixed
/Flex_vol_Test01     Test_dest_flexpolicy Flex_vol_Test01 volume 2  read-write mixed

2 entries were displayed.
```

```
ONTAPSelectCluster::> export-policy check-access -vserver SVM_Dataflex -volume Flex_vol_Test01 -client-ip 10.12.96.8 -authentication
method sys -protocol nfs3 -access-type read-write
                            Policy    Policy     Rule            Security
Path                Policy  Owner     Owner Type Index Access    Style
------------------- ------- --------- ---------- ---- --------- ---------
/                   default SVM_Dataflex_root volume 2 read      mixed
/Flex_vol_Test01    Test_dest_flexpolicy Flex_vol_Test01 volume 2 read-write mixed
2 entries were displayed.
```

Expected Output

- Checks access for client IP 10.12.96.8 on volume **Flex_vol_Test01**
- Shows **Vserver:** SVM_dataflex
- Authentication method: **sys**
- Protocol: **NFSv3**
- Access type tested: **read-write**
- Output indicates whether access is **allowed** or **denied**
- Confirms which **export policy rule** permits or blocks the access

Test Mount on Vultr Compute

Verification that Vultr compute nodes can access the replicated dataset over NFS. Ensures the end-to-end path - from **on-prem** » **NetApp FlexCache** » **NetApp ONTAP(Destination)** » **Vultr compute** - is fully operational.

When we mount an NFS export from NetApp ONTAP(Destination) in Vultr:

- We are mounting the **active filesystem** of the destination NetApp FlexCache Volume
- NetApp FlexCache maintains a local cache of hot (frequently accessed) data from the source volume. Clients can perform read and write operations on the NetApp FlexCache volume—reads are served locally from the cache, while all writes are sent directly to the origin (source) volume, which remains the authoritative data source.

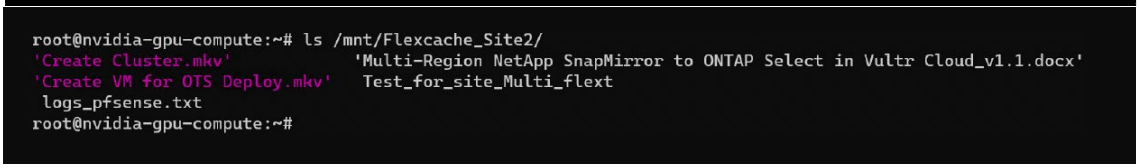
CLI (from compute instance on Vultr Cloud)

```
mount -t nfs <ontap_select_ip>:/<junction-path> <compute node mount point>
```

```
root@nvidia-gpu-compute:/mnt/Flexcache_Site2# mount -t nfs 10.12.96.53:/Flex_vol_Test01 /mnt/Flexcache_Site2
```


```
ls /mnt/Flexcache_Site2/
```

```
root@nvidia-gpu-compute:~# ls /mnt/Flexcache_Site2/
'Create Cluster.mkv'            'Multi-Region NetApp SnapMirror to ONTAP Select
                                 in Vultr Cloud_v1.1.docx'
'Create VM for OTS Deploy.mkv'   Test_for_site_Multi_flext
logs_pfsense.txt
root@nvidia-gpu-compute:~#
```



Testing Read Access

Once the FlexCache volume is mounted, we can verify that the end user is able to successfully access and read the cached data from the cache volume.

```
cat /mnt/Flexcache_Test/<file>
```

```
root@nvidia-gpu-compute:/mnt/Flexcache_Site2# cat logs_pfsense.txt
Last 500 IPsec Log Entries. (Maximum 500)
Nov 28 09:08:55 14[CFG] remote_port = 500
Nov 28 09:08:55 14[CFG] send_certreq = 1
Nov 28 09:08:55 14[CFG] send_cert = CERT_SEND_IF_ASKED
Nov 28 09:08:55 14[CFG] ppk_id = (null)
Nov 28 09:08:55 14[CFG] ppk_required = 0
Nov 28 09:08:55 14[CFG] mobike = 0
Nov 28 09:08:55 14[CFG] aggressive = 0
Nov 28 09:08:55 14[CFG] dscp = 0x00
Nov 28 09:08:55 14[CFG] encap = 0
```


```
root@nvidia-gpu-compute:/mnt/Flexcache_Site2# cat logs_pfsense.txt
Last 500 IPsec Log Entries. (Maximum 500)
Nov 28 09:08:55 14[CFG]         remote_port = 500
Nov 28 09:08:55 14[CFG]         send_certreq = 1
Nov 28 09:08:55 14[CFG]         send_cert = CERT_SEND_IF_ASKED
Nov 28 09:08:55 14[CFG]         ppk_id = (null)
Nov 28 09:08:55 14[CFG]         ppk_required = 0
Nov 28 09:08:55 14[CFG]         mobike = 0
Nov 28 09:08:55 14[CFG]         aggressive = 0
Nov 28 09:08:55 14[CFG]         dscp = 0x00
Nov 28 09:08:55 14[CFG]         encap = 0
```


Testing Write Access (Destination NetApp ONTAP)

The following commands demonstrate the testing of FlexCache write functionality. A file named **Testing_for_FlexCache_Write** was successfully created from the cache volume, and the same file is now available and visible on the source NFS host, confirming that write operations are correctly redirected to the source volume.

```
root@nvidia-gpu-compute:/mnt/Flexcache_Site2# touch Test_for_site_Multi_flext-Region-B
root@nvidia-gpu-compute:/mnt/Flexcache_Site2#
root@nvidia-gpu-compute:/mnt/Flexcache_Site2# ls -l
total 1225500
-rw-r--r-- 1 nobody nogroup 681090009   Dec 16 10:56  'Create Cluster.mkv'
-rw-r--r-- 1 nobody nogroup 567048214   Dec 16 10:56  'Create VM for OTS Deploy.mkv'
-rw-r--r-- 1 nobody nogroup     27476   Dec 15 10:00  logs_pfsense.txt
-rw-r--r-- 1 nobody nogroup   1803092   Dec 15 10:00  'Multi-Region NetApp SnapMirror to ONTAP
Select in Vultr Cloud_v1.1.docx'
-rw-r--r-- 1 nobody nogroup         0   Dec 15 11:23  Test_for_site_Multi_flext
-rw-r--r-- 1 nobody nogroup         0    Feb 4 09:24  Test_for_site_Multi_flext-Region-B
root@nvidia-gpu-compute:/mnt/Flexcache_Site2#
```


```
root@nvidia-gpu-compute:/mnt/Flexcache_Site2# touch Test_for_site_Multi_flext-Region-B
root@nvidia-gpu-compute:/mnt/Flexcache_Site2#
root@nvidia-gpu-compute:/mnt/Flexcache_Site2# ls -l
total 1225500
-rw-r--r-- 1 nobody nogroup 681090009 Dec 16 10:56 'Create Cluster.mkv'
-rw-r--r-- 1 nobody nogroup 567048214 Dec 16 10:56 'Create VM for OTS Deploy.mkv'
-rw-r--r-- 1 nobody nogroup     27476 Dec 15 10:00  logs_pfsense.txt
-rw-r--r-- 1 nobody nogroup   1803092 Dec 15 10:00 'Multi-Region NetApp SnapMirror to ONTAP Select in Vultr Cloud_v1.1.docx'
-rw-r--r-- 1 nobody nogroup         0 Dec 15 11:23  Test_for_site_Multi_flext
-rw-r--r-- 1 nobody nogroup         0 Feb  4 09:24  Test_for_site_Multi_flext-Region-B
root@nvidia-gpu-compute:/mnt/Flexcache_Site2#
```


Testing Write Through to Origin (Source NetApp ONTAP)

ONTAP FlexCache uses **write-through semantics**, where all writes are forwarded directly to the origin volume, preserving a single authoritative source of truth. A write or update issued at the **FlexCache mount** is **forwarded to the origin volume.**


```
ubuntu@ip-172-31-12-99:/mnt/Flex_Vol_S20$ ls -l
total 1225500
-rw-r--r-- 1 root root 681090009 Dec 16 10:56 'Create Cluster.mkv'
-rw-r--r-- 1 root root 567048214 Dec 16 10:56 'Create VM for OTS Deploy.mkv'
-rw-r--r-- 1 root root   1803092 Dec 15 10:00 'Multi-Region NetApp SnapMirror to ONTAP Select
in Vultr Cloud_v1.1.docx'
-rw-r--r-- 1 root root         0 Dec 15 11:23  Test_for_site_Multi_flext
-rw-r--r-- 1 root root         0  Feb 4 09:24  Test_for_site_Multi_flext-Region-B
-rw-r--r-- 1 root root     27476 Dec 15 10:00  logs_pfsense.txt
ubuntu@ip-172-31-12-99:/mnt/Flex_Vol_S20$
```


```
ubuntu@ip-172-31-12-99:/mnt/Flex_Vol_S20$ ls -l
total 1225500
-rw-r--r-- 1 root root 681090009 Dec 16 10:56 'Create Cluster.mkv'
-rw-r--r-- 1 root root 567048214 Dec 16 10:56 'Create VM for OTS Deploy.mkv'
-rw-r--r-- 1 root root   1803092 Dec 15 10:00 'Multi-Region NetApp SnapMirror to ONTAP Select in Vultr Cloud_v1.1.docx'
-rw-r--r-- 1 root root         0 Dec 15 11:23  Test_for_site_Multi_flext
-rw-r--r-- 1 root root         0 Feb  4 09:24  Test_for_site_Multi_flext-Region-B
-rw-r--r-- 1 root root     27476 Dec 15 10:00  logs_pfsense.txt
ubuntu@ip-172-31-12-99:/mnt/Flex_Vol_S20$
```


Expected Output

- Mount is successful
- Read and Write operations success

Validate Throughput and Performance

A performance test using read workloads to ensure ONTAP meets expected throughput for AI/analytics. Confirms the system can sustain the I/O patterns required by downstream GPU/compute pipelines.

The dd test is run twice to demonstrate FlexCache behavior: the first run reads data from the origin across the WAN, while the second run reads the same data locally from the FlexCache, delivering higher throughput.

CLI (compute-side example)

```
dd if=/mnt/Flexcache_Site2/<bigfile> of=/dev/null bs=1M status=progress
```

```
root@nvidia-gpu-compute:/mnt/Flexcache_Site2# dd if=cluster1.mkv of=/dev/null bs=1M status=progress
677380096 bytes (677 MB, 646 MiB) copied, 19 s, 35.6 MB/s
649+1 records in
649+1 records out
681090009 bytes (681 MB, 650 MiB) copied, 19.1285 s, 35.6 MB/s
```

Expected Output (from across the WAN)

- Stable read throughput (e.g., **20–200** MB/s depending on network access to the source over the WAN)
- No read stalls or NFS timeout messages

Now clear linux cache (so test doesn't take data from memory but from FlexCache)

```
echo 3 > /proc/sys/vm/drop_caches
```

```
root@nvidia-gpu-compute:/mnt/Flexcache_Site2# dd if=cluster1.mkv of=/dev/null bs=1M status=progress
668991488 bytes (669 MB, 638 MiB) copied, 2 s, 334 MB/s
649+1 records in
649+1 records out
681090009 bytes (681 MB, 650 MiB) copied, 2.03747 s, 334 MB/s
```

Expected Output (from FlexCache)

- Stable read throughput (e.g., **200–400** MB/s)
- No read stalls or NFS timeout messages

Observation: Higher Performance Gains with FlexCache

Accessing data through ONTAP FlexCache can deliver significantly higher throughput than reading directly from the origin volume over the WAN, because frequently accessed (hot) data is served from the FlexCache volume that is mounted within the same region or site as the client. Although FlexCache is accessed over the network, it avoids WAN round trips by servicing read requests from a nearby cache, reducing latency and offloading read traffic from the origin volume. This enables near–local-storage performance for repeated read operations while preserving a single authoritative source of truth at the origin.

Troubleshooting

A structured approach to isolate problems across networking, routing, NetApp FlexcaCheе, and export access.

This section covers

- Peering problems, including failed intercluster or SVM peer relationships
- NetApp FlexCache transfer failures caused by policy, scheduling, permissions, or connectivity issues
- Routing or firewall misconfigurations that block NetaApp FlexCache ports or intercluster LIF communication
- ONTAP connectivity issues related to VPC, VPN, or LIF configuration errors
- NFS/SMB export issues impacting read-only mount access from compute nodes

Validate Intercluster LIF Connectivity

From source cluster:

```
network ping -lif <source_intercluster_lif> -vserver <src_svm> -destination <dest_
intercluster_lif>
```

Expected:

- Successful ICMP replies
- Low latency for smooth transfers

Validate SVM and Cluster Peering

Cluster peering:

```
cluster peer show
```

Look for:

- Availability = Available
- Authentication Status = ok

SVM peering:

```
vserver peer show
```

Look for:

- Peer state = **peered**

Expected:

- Peer state = peered
- Applications = snapmirror , flexcache
- No conflicts or pending states

Detailed peer status

```
cluster peer show -instance
```

Expected:

- No blocked ports

- No authentication failures
- Connection status = operational

Routing or Firewall Misconfiguration

Ping intercluster LIFs

```
network ping -lif <src_ic_lif> -vserver <src_svm> -destination <dst_ic_lif>
```

Expected:
- Successful ping replies
- Low latency (single digit ms ideal)

Routing table

```
network route show
```

Expected:
- Route exists to destination intercluster network
- Gateway reachable
- No incorrect or overlapping routes

Firewall policy check

```
system services firewall policy show
```

Expected:
- FlexCache service allowed
- No deny rules blocking intercluster traffic
- Relevant policies applied to the correct LIFs

Verify NetApp FlexCache ports

```
network connections active show -service fcache
```

Expected:
- Active connections visible on ports 11104 and 11105
- State = Established

ONTAP Connectivity (VPC/VPN/LIF)

List all LIFs

```
network interface show
```

Expected:
- All LIFs in up/up state
- Correct roles: data / intercluster / mgmt

Home-node and home-port correctness

```
network interface show -fields home-node,home-port,is-home
```

Expected:
- is-home = true (unless failover active)

Cluster internal connectivity

```
cluster ping-cluster -node *
```

Expected:

- All nodes reachable
- No packet loss

Test VPN reachability

```
network ping -vserver <svm> -destination <onprem_vpn_gateway_ip>
```

Expected:

- Ping successful
- Round-trip latency consistent

NFS Export Issues

Export policies

```
vserver export-policy rule show
```

Expected:

- Client IP or subnet allowed
- Read-only permissions granted
- NFS version allowed (e.g., v3/v4)

Volume export path

```
volume show -fields junction-path,export-policy
```

Expected:

- Junction path = valid (e.g., /vol1)
- Export policy assigned correctly

List CIFS shares

```
vserver cifs share show
```

Expected:

- Share listed (if SMB used)
- Correct permissions

Check client access

```
export-policy check-access -vserver <svm> -volume <vol> -client-ip <compute_ip>
-authentication-method sys
```

Expected:

- Access = read or read-only
- No “Access denied”

NFS service status

```
vserver nfs status
```

Expected:

- NFS enabled = true
- Active NFS versions listed

FlexCache Volume Issues

Volume Status

```
volume show –volume <cache_volume>
```

Expected :

- Displays details of the specified **FlexCache volume**
- Shows **volume name** and associated **Vserver (SVM)**
- Indicates volume state (e.g., online)
- Displays **volume size** and **available space**
- Shows **junction path** (if mounted)
- Confirms the volume is a **FlexCache** type and linked to an origin volume

Flexvolume status :

```
volume flexcache show
```

Expected:

- Lists all FlexCache volumes in the cluster
- Displays cache volume name and Vserver
- Shows associated **origin volume** and **origin Vserver**
- Indicates cache status as **available/online**
- Displays cache size and usage
- Confirms FlexCache relationship is healthy

Origin Volume status:

```
volume flexcache origin show
```

Expected :

- Displays origin volumes enabled for FlexCache
- Shows origin Vserver and volume name
- Lists associated cache volumes
- Confirms FlexCache origin status
- Indicates active cache relationships